Веселина Спасова

СИГУРНОСТ ПО ДИЗАЙН В СОФТУЕРНОТО ИНЖЕНЕРСТВО

ВСУ „Черноризец Храбър“
Издателски център
2022

ISBN 978-954-715-834-7

Познаването и прилагането на стандартите за управление на информационната сигурност и принципите за сигурност по дизайн има за фирмите от софтуерния бизнес двойно значение и приложение.

На първо място, като безнес организации, те могат да внедрят собствена система за управление на информационната сигурност, като това би им донесло редица предимства. От особено голямо значение би било сертифицирането за компании, занимаващи се с ИТ услуги, обработващи голямо количество данни на потребителите: поддръжка на платформи за електронна търговия, фирми, осъществяващи дигитален маркетинг и др. Както вече споменахме, сертифицирането е задължително за компаниите, осъществяващи ИТ услуги в публичния сектор и е условие за участие на фирмата в обществени конкурси за разработка на софтуер за такива услуги.

На второ място, като компании, разработващи и предлагащи софтуер и ИТ услуги на бизнес клиенти, те, като част от веригата за доставки, трябва да гарантират спазването на изискванията за информационна сигурност на сертифицирани клиенти и да докажат че техния продукт няма да създаде условия за пробиви в системите на клиента.

Считам, че настоящата монография ще бъде полезна и в двете насоки.

СЪДЪРЖАНИЕ

Първа глава
ПОЛИТИКИ И СТАНДАРТИ ЗА ИНФОРМАЦИОННА СИГУРНОСТ

Втора глава
СТАНДАРТИ ЗА ИНФОРМАЦИОННА СИГУРНОСТ

Трета глава
СИГУРНОСТ ПО ДИЗАЙН

Четвърта глава
НАЦИОНАЛНИ ПОЛИТИКИ И ПРИНЦИПИ
ЗА ПРИЛАГАНЕ НА СИГУРНОСТ ПО ДИЗАЙН
ПРИ РАЗРАБОТКА НА СОФТУЕРНИ СИСТЕМИ
77

Първа глава
ПОЛИТИКИ И СТАНДАРТИ ЗА ИНФОРМАЦИОННА СИГУРНОСТ

1. Същност на информационната сигурност

Киберсигурността е все по-важен а за някои организации и стратегически елемент за тяхната дейност, от който зависи не само престижа им а в някои случаи и възможността за функциониране или даже оцеляване. Все повече организации обръщат внимание на различните аспекти на сигурността на данните и информацията в различните й състояния и докато физическите методи за охрана и опазване имат дълга история в годините, то развитието на информационните технологии в тяхната сложност и всеобхватност създава нови предизвикателства пред бизнеса и организациите по защита на данните и информацията от една страна и пред разработчиците на софтуер да отговорят на тези изисквания от друга. Въпреки, че в специализираната литература въпросът за сигурността на информацията се разглежда отдавна (Сребров, 1989), (Арнаудов, 2007), то през последните години прави впечатление засиленото внимание на обществото и институциите към този проблем и появата на редица нормативни актове и стандарти, които правят опити за неговото разрешаване и превръщане в политика и ежедневна практика.

Въпросите за същността на данните и информацията и тяхното точно определение са едни от най-дискутираните в компютърните науки и могат да бъдат намерени десетки различни определения. Все пак ще се придържаме към общоприетото разбиране, че данните са отделни факти за различни обекти или събития от реалния свят, докато информацията представлява данни, които са събрани и обработени с определена цел (Пенева, 2004).

Един от важните аспекти на киберсигурността са нейните правни основи. В рамките на Европейския съюз, НАТО и на държавно ниво съществуват редица актове, част от които имат задължителен, други препоръчителен характер, които трябва да се имат

предвид. Тази материя е подробно разгледана в книгата „Правни основи на сигурността“ на Драгомир Кръстев (Кръстев, 2021) и не е предмет на тази книга, освен в частите си, които имат директно отношение към процесите на проектиране, разработка и експлоатация на софтуерните системи.

По отношение на понятията, свързани с киберсигурността, а именно киберсигурност и информационна сигурност ще използваме определенията, дадени в национални и международни документи и стандарти. Определението за киберсигурност, дадено в речника към Националната стратегия за киберсигурност „Киберустойчива България 2023“, последно актуализирана 2021 гласи „състояние на обществото и държавата, при което чрез прилагане на комплекс от мерки и действия в киберпространството е защитено от заплахи, свързани с неговите независими мрежи и информационна инфраструктура или които могат да нарушат работата им. Киберсигурността включва мрежова и информационна сигурност, противодействие на киберпрестъпността и кибер отбрана (Съгласно чл. 2, ал. 1 и ал. 2, на ЗКС)“. Самото определени съдържа в себе си два вида сигурност: мрежова и информационна. Доколкото тази книга е посветена на осигуряване на адекватно ниво на сигурност в процеса на проектиране на софтуерни системи и тяхната експлоатация, то ще се фокусираме върху втория елемент на киберсигурността, а именно – информационната сигурност.

В нормативните документи обаче не се съдържат самостоятелни определения на тези понятия. Обхвата на информационната сигурност може да бъде извлечен от определението за мрежова и информационна сигурност от закона за киберсигурност и косвено от определенията за киберпрестъпност, кибератака, киберинцидент и риск. За да дефинира същността на тези понятия самата стратегия цитира няколко документа, като основните са стандартите от серията ISO 27000 за информационна сигурност, важни аспекти на който ще разгледаме по-късно, както и стратегически документи на НАТО и Европейския съюз. Ще цитираме точно дадените формулировки, за да можем да извлечем от тях различните аспекти на информационната сигурност.

Киберпрестъпност (ЕС) – обхваща традиционни престъпления (например измами, фалшифициране и кражба на самоличност), престъпления, свързани със съдържанието (напр. онлайн разпространение на детска порнография или подбуждане към расова омраза), и престъпления, които са възможни само при компютри и информационни системи (например атаки срещу информационни системи, предизвикване на отказ на услуга и зловреден софтуер).

Кибератака – опит за разрушаване, разкриване, променяне, забрана, кражба или получаване на неупълномощен достъп до/или неупълномощено използване на информационен актив. (Съгласно т. 10 от ДР на ЗКС)

(НАТО) – действия, предприети за нарушаване, отхвърляне, влошаване или разрушаване на информация, намираща се в компютър и/или компютърна мрежа, както и на самите компютри и/или компютърни мрежи.

(ISO 27000) – опит за разрушаване, разкриване, променяне, забрана, кражба ли получаване на неупълномощен достъп до или реализация на неупълномощено използване на актив.

Киберинцидент – събитие или поредица от нежелани или неочаквани събития, свързани с киберсигурността, които с голяма вероятност могат да предизвикат компрометиране на дейностите и заплашват сигурността на информация. (Съгласно т. 12 от ДР на ЗКС)

(НАТО) – неочаквано събитие в киберпространството, което, с или без криминален умисъл, би могло да промени киберсигурността чрез фактическо или потенциално излагане на опасност на конфиденциалността, целостта или наличността на информационната система или на информацията, която системата обработва, съхранява или пренася, нарушаване или потенциално нарушаване на политиките за сигурност, процедурите за сигурност или политиките за приемливо използване.

(ISO 27000) – събитие или поредица от нежелани или неочаквани събития, свързани със киберсигурността, които с голяма вероятност могат да предизвикат компрометиране на дейностите и заплашват сигурността на информацията.

Критична инфраструктура – стратегически обекти и дейности от значение за националната сигурност – Съгласно чл. 1, ал. 1 на ПМС№181 от 20 юли 2009 г., стратегическите обекти и дейности, които от значение за националната сигурност, се определят в единен списък и са част от критичната инфраструктура. Законово определение на КИ се съдържа в §1, т. 15 на ДР на Закона за защита при бедствия: „*Система или части от нея, които са от основно значение за поддържането на жизненоважни обществени функции, здравето, безопасността, сигурността, икономическото или социалното благосъстояние на населението и чието нарушаване или унищожаване би имало значителни негативни последици за Република България в резултат на невъзможността да се запазят тези функции*“.

Критична комуникационна и информационна инфраструктура – системи, услуги, мрежи и инфраструктури, които са жизнено важна част от националната икономика и общество и осигуряващи важни стоки и услуги, деструктивното въздействие върху които би могло да има сериозно влияние на жизненоважни функции на обществото. Критична информационна инфраструктура са както мрежите, каналите и системите за управлението и поддържането им.

Мрежова и информационна сигурност – способност на мрежите и информационните системи да се противопоставят на определено ниво на въздействие, засягащи отрицателно наличието, истинността, целостта, или поверителността на съхранявани, пренасяни или обработвани данни или на свързаните с тях услуги, предлагани от тези мрежи и информационни системни или достъпни чрез тях. (Съгласно чл. 2, ал. 3 на ЗКС).

Няколко са изводите, които може да бъдат направени на базата на тези определения:

1. Интензификацията на дигитализацията във всички сфери оказва съществено влияние на критичната инфраструктура. Доколкото тази инфраструктура включва не само хардуера а и системите и услугите, поддържащи жизнено важни за обществото функции, като например здравеопазване, финанси и др., то при осъществя-

ване на процесите по дигитализация трябва да се обръща особено внимание на оценката на риска от загуба на данни и невъзможност за предоставяне на услуга и да бъдат предвидени мерки за тяхното предотвратяване;

2. Системите трябва да поддържат функционалност за проследяване на действията на операторите, откриване, навременно информиране и при възможност предотвратяване на инциденти. Това е свързано и с възможностите за разследване от компетентните органи и доказване на наказателна отговорност от извършителя в случай на извършване на деяние, което може да бъде квалифицирано като киберпрестъпление.

3. Като киберинцидент трябва да се класифицира не само сбъднало се събитие, довело до компрометиране на информацията а и възможността това да се случи, т.е. пропуските в осигуряване на сигурността на информацията, допуснати на етапите по проектиране или разработка, дори и те да не са използвани за атака. Или, с други думи, редно е при класифицирането на бъговете в една софтуерна система да отделим специална група, които биха повлияли на сигурността и да наречем тази група „инцидентни бъгове“;

4. Не можем да очакваме разработените системи да работят безотказно във всякакви ситуации, но е важно много точно да определим нивото на въздействие, които могат да понесат преди критичен срив и още повече, още в процеса на проектиране да заложим нивото на сигурност, които искаме да постигнем и да определим критерии за оценка, както в момента на въвеждане на системата в експлоатация, така и по-късно в процеса на нейната поддръжка, доколкото променящите се условия на средата могат да променят, като най-четно това означава да понижат това ниво. Следователно, още в началото трябва да бъдат въведени и да бъдат осигурени инструменти за измерване и наблюдение на показатели за нивото на сигурност на системата през целия жизнен цикъл.

5. Сигурността на информацията се характеризира с четири основни аспекта: наличност, истинност, цялост и поверителност, като тези аспекти имат своите проекции в три основни процеса:

съхраняване, пренасяне и обработка, т.е. всеки от аспектите трябва да бъде оценен за всеки от процесите. Трябва да отбележим, че стандартите от серията ISO 27000 дефинират само 3 аспекта, 3 стълба: конфиденциалност, интегритет и наличност (CIA – **C**onfidentiality, **I**ntegrity, **A**vailability). Впечатление прави, че българското законодателство въвежда допълнителна характеристика – истинност на информацията, като от изключителен интерес представлява начина, по който тази характеристика ще бъде измервана и оценявана във времето. Някои насоки за това се съдържат в Регламент 2016/679 на ЕС, добил популярност като регламент за защита на личните данни.

В заключение на базата на множеството цитирани определения можем да дефинираме информационната сигурност като способност на информацията да бъде налична, истинна, цялостна и поверителна в процесите на нейното съхранение, пренос и обработка до определено ниво на въздействие.

Конфиденциалността е способността информацията да не се предоставя или разкрива на неупълномощени лица, или процеси.

Цялост (интегритет) – свойство на точност и пълнота на информацията.

Наличност – свойство информацията да бъде достъпна и използваема при поискване от упълномощен субект.

Освен това могат да бъдат включени и други свойства, като автентичност, отчетност, неотрицаемост и надеждност.

От тези определения за информационната сигурност и нейните свойства до голяма степен произтичат и действията, които трябва да бъдат предприети в процеса на инженеринг на софтуерните проекти.

2. Сигурност, базирана на оценката на риска

Сигурността не може да бъде абсолютна, тя винаги е по отношение на дадени обекти и се отнася за дадени ситуации. Винаги съществува възможност за възникване на нови, непредвидени ситуации, за които системите на се защитени. За да можем в максимална степен да прогнозираме и оценим възможните опасни ситуации трябва да сме наясно с механизма на заплахата и влия-

нието ѝ върху активите на организацията. Активите може да бъдат материални, нематериални и човешки, като обект на разглеждане в тази публикация са атаки само срещу нематериални активи на организацията, съхранявани в електронен вид в рамките на нейната ИКТ инфраструктура или наети ресурси, осигуряващи бизнес процеси, изпълнявани в рамките на организацията. Въпреки това условно разграничаване, всички елементи трябва да бъдат разглеждани като система, тъй като атаката се осъществява най-често посредством достъп чрез комуникационната система и понякога и със съзнателното или несъзнателно участие на персонала.

Заплахата е потенциал за нарушаване на сигурността, който съществува, когато има обстоятелство, възможност, действие или събитие, което може да наруши сигурността и да причини вреда (IEC Guide 120).

Заплахите се материализират като атаки или опасни събития. **Атаката** е нападение срещу система, което произтича от интелигентна заплаха – т.е. интелигентно действие, което е умишлен опит (особено в смисъла на метод или техника) за избягване на услугите за сигурност и нарушаване на политиката за сигурност на системата (Ръководство на IEC 120 , 3.2).

Заплахите и атаките са възможни и осъществими, защото даден актив може да има уязвимости в сигурността. **Уязвимостта** е недостатък или слабост в дизайна, внедряването или работата и управлението на системата, които могат да бъдат използвани за нарушаване на политиката за сигурност на системата.

Всяка атака нанася щети на даден актив. Щетата се дефинира като нараняване или увреждане на здравето на хората, увреждане на активи или околната среда (ISO/IEC Ръководство 51, 3.1). Враждебните действия или влияния могат да бъдат умишлени или неумишлени. От своя страна сигурността е състояние, което е резултат от установяването и поддържането на защитни мерки, които гарантират състояние на неприкосновеност от враждебни действия или влияния (Ръководство на IEC 120, 3.13).

Рискът е относителна мярка за степента, до която даден актив е застрашен от потенциално обстоятелство. Рискът се дефи-

нира като комбинация от вероятността за възникване на вреда и тежестта на тази вреда (IEC Guide 120, 3.11). Рисковете за информационната сигурност са онези рискове, които произтичат от загуба на поверителност, цялост или наличност на информация или информационни системи и отразяват потенциалните неблагоприятни въздействия върху организационните операции (т.е. мисия, функции, имидж или репутация), организационните активи, лица, други организации и нацията (NIST SP 800-30). От своя страна процесът на оценка на риска е процес на идентифициране, оценка и приоритизиране на рисковете за информационната сигурност (NIST Special Publication 800-30 Revision 1).

Моделите на риска определят рисковите фактори, които трябва да бъдат оценени, и връзките между тези фактори. Рисковите фактори са характеристики, използвани в моделите на риска като входни данни за определяне на нивата на риск при оценките на риска.

3. Модели за оценка на риска

Европейската агенция по киберсигурност (ENISA – EU Agency for Cybersecurity) публикува през януари 2022 година доклада „Оперативна съвместимост на рамките за управление на риска“ (Interoperable EU Risk Management Framework, 2021).

ISO/IEC 2382 дефинира оперативната съвместимост като способността за комуникация, изпълнение на програми или прехвърляне на данни между различни функционални единици по начин, който изисква от потребителя да има малко или никакви познания за уникалните характеристики на тези единици.

Стандартният компютърен речник на IEEE също поставя акцент върху необходимото усилие, като по този начин определя оперативната съвместимост като способността на система или продукт да работи с други системи или продукти без специални усилия от страна на клиента. Това определение е прието и от ISO 23903 по отношение на оперативната съвместимост в здравния сектор.

Европейската рамка за оперативна съвместимост определя оперативната съвместимост като „способността на организаци-

ите да си взаимодействат за постигане на взаимноизгодни цели, включващи споделяне на информация и знания между тези организации, чрез бизнес процесите, които те поддържат, посредством обмен на данни между техните ИКТ системи.

Като се имат предвид горните дефиниции, както и структурата на рамките за управление на кибер рисковете и целите на техните индивидуални функционални характеристики, оперативната съвместимост на рамките и методологиите за управление на риска може да се определи като способността на компонент или методи за управление на риска да използват повторно информацията, предоставена от компонентите или методите за управление на риска на други рамки със същата лекота и със същите интерфейси, към същите цели.

Този доклад предлага методология за оценка на потенциалната оперативна съвместимост на рамките и методологиите за управление на риска (RM) и представя свързаните с нея резултати. Методологията, използвана за оценка на оперативната съвместимост, произтича от задълбочено проучване на литературата, което води до използването на някои характеристики на рамката на RM, които са избрани за тази цел. Тези характеристики, които са идентифицирани като подходящи за оценката на оперативната съвместимост, са подробно описани и анализирани за всяка рамка/методология. По-конкретно, за определени функционални характеристики е използвана скала от четири нива, за да се оцени нивото на оперативна съвместимост за всеки метод и всеки набор от комбинирани функции. Разгледани са общо 16 методологии/рамки, като десет от тях са базирани на активи, три се считат за базирани на сценарии, а останалите три носят характеристики, както на базирани на активи, така и на базирани на сценарии. По подобен начин подмножеството от методи за анализ включва както количествени (само 2 от 16), така и качествени (10 от 16) методологии, докато 4 от тях имат характеристиките и на двете категории. По отношение на потенциалната оперативна съвместимост на анализираните методи, всички те изглеждат силно оперативно съвместими по отношение на заплахи и мерки, което позволява приемането на допълнителни каталози, предоставени

от други методи, или промяната на съществуващите. Три от анализираните методологии не отчитат уязвимостите в своя подход към оценката на риска.

Освен това 9 от 16-те методологии се считат за силно оперативно съвместими по отношение на техния подход за изчисляване на риска и следователно са по-отворени за приемане на алтернативи, докато 7 от 16-те методологии позволяват модификация на предложения метод за изчисляване на риска, обикновено по отношение на везните, които се използват. Нивата на оперативна съвместимост на анализираните методологии са обобщени и представени в таблица, в която са представени оценките по всички показатели и позволява на всяка организация да избере най-подходящата за нея.

3.1. CRAMM (CCTA Risk Analysis and Management Method)

Въпреки, че не присъства в доклада, се счита за най-простия метод за анализ на риска, разработен от британската правителствена организация CCTA (Централна агенция за комуникация и телекомуникации), сега преименувана на Служба за правителствена търговия (Office of Government Commerce – OGC). Инструмент със същото име поддържа метода: CRAMM. В момента е актуална неговата пета версия. Методът CRAMM е доста труден за използване без инструмента CRAMM. Първите версии на CRAMM (метод и инструмент) се основават на най-добрите практики на британски правителствени организации. Понастоящем CRAMM е предпочитаният метод за анализ на риска от правителството на Обединеното кралство, но се използва и в много страни извън Обединеното кралство – НАТО, холандските въоръжени сили и корпорации, работещи активно по сигурността, като Unisys. Той е особено подходящ за големи организации, като държавни органи и индустрия. Повече информация за метода може да бъде намерена в бялата книга, посветена на него, достъпна на адрес https://www.sans.org/white-papers/83/ (Yazar, 2021).

3.2. Подход NIST SP 800-30

Целта на Специална публикация 800-30 на Националния институт за стандарти и технологии (NIST – National Institute of

Standards and Technology, U.S. Department of Commers) е да предостави насоки за извършване на оценки на риска на федерални информационни системи и организации, разширявайки насоките в Специална публикация 800-39. Оценките на риска, извършени и на трите нива в йерархията за управление на риска, са част от цялостен процес на управление на риска – предоставяйки на висшите ръководители/изпълнители информацията, необходима за определяне на подходящи курсове на действие в отговор на идентифицираните рискове.

NIST SP 800-30 се използва за извършване на оценка на риска в рамките на параметрите на рамката на NIST за идентифициране, оценка и приоритизиране на риска за работата на организациите.

Целта на оценката на риска е да информира лицата, вземащи решения, и да подкрепи техните реакции на риска, като им предостави информация за:

- вътрешни и външни уязвимости;
- съответни заплахи за организацията;
- въздействие върху организацията
- вероятност от настъпване на вреда

Това, в крайна сметка води до определяне на рисковете.

Оценките на риска са част от процеса на управление на риска. Тези оценки се извършват на трите нива на йерархията за управление на риска, от своя страна, предоставяйки дълбока представа на висшето ръководство, за да предприеме необходимите действия в отговор на идентифицираните рискове.

По-специално, ръководствата на SP 800-30 изпълняват следните стъпки от процеса на оценка на риска:

- подготовка за оценка на риска;
- провеждане на оценката;
- съобщаване на резултатите от оценката и поддържането му.

Специална публикация 800-30 също ръководи за идентифициране на специфични рискове, които да се наблюдават постоянно (Това помага на организацията да определи дали рисковете са надхвърлили прагова стойност, която е извън допустимия риск на организацията) и различни курсове на действие, които да бъдат предприети.

Типичните рискови фактори включват:

- заплаха (или източник на заплаха),
- атака (или заплаха),
- уязвимости,
- ниво на неблагоприятно въздействие (или тежест на вредата)
- вероятност (или възникване на вреда) и
- предразполагащо състояние.

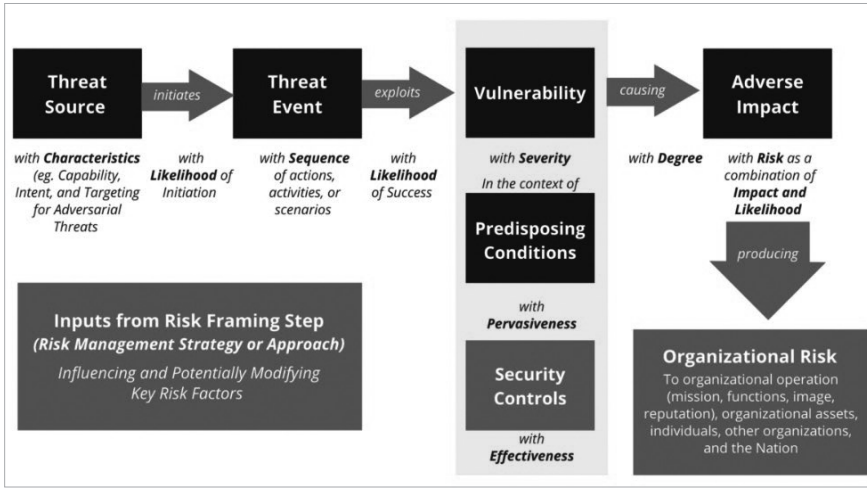
Всяка вреда се оценява според тежестта на вредата или неблагоприятното въздействие. Неблагоприятното въздействие от атака е степента на вредата, която може да се очаква да произтече от последствията от неоторизирано разкриване на информация, неоторизирана модификация на информация, неоторизирано унищожаване на информация или загуба на информация или наличност на информационна система.

Въздействието е положителна или отрицателна промяна в обществото, икономиката или околната среда, изцяло или частично в резултат на минали и настоящи решения и дейности на организацията (ISO 26000:2010, 2.9)

Появата на вреда е претеглен рисков фактор, базиран на анализ на вероятността дадена заплаха да може да използва дадена уязвимост (или набор от уязвимости) (Андреев, 2021).

В допълнение към уязвимостите, организациите вземат предвид и предразполагащите условия. Предразполагащо състояние е състояние, което съществува в организация, мисия или бизнес процес, корпоративна архитектура, информационна система или работна среда, което влияе (т.е. увеличава или намалява) вероятността заплахите, веднъж инициирани, да доведат до неблагоприятни въздействия върху организационните операции и активи, лица или други организации (Петрова, 2021).

Предразполагащите условия включват, например, местоположението на съоръжение в район, предразположен към урагани или наводнения (увеличава вероятността от излагане на урагани или наводнения) или самостоятелна информационна система без свързаност с външна мрежа (намалява вероятността от излагане до мрежова кибератака).



Фигура 1. Аналитичен подход и модел на риска (Специална публикация на NIST 800-30 Ревизия 1, фиг. 3)

Важно е, също така, да се отбележи, разграничаването на понятията сигурност и безопасност. Това може най-лесно да бъде илюстрирано чрез пример. Ако разгледаме един автомобил, то автомобилната сигурност се отнася до защита на колата и нейното съдържание от престъпна дейност.

Безопасността на автомобила е свързана с защитата на хората, като намалява вероятността автомобилът да участва в произшествие и включва функции, които означават, че е по-малко вероятно хората да бъдат наранени, ако има инцидент. На практика безопасност та е свобода от риск, който е непоносим. При преводите обаче на стандартите от английски на български език обаче, този нюанс е загубен и двете понятия не са ясно разграничени.

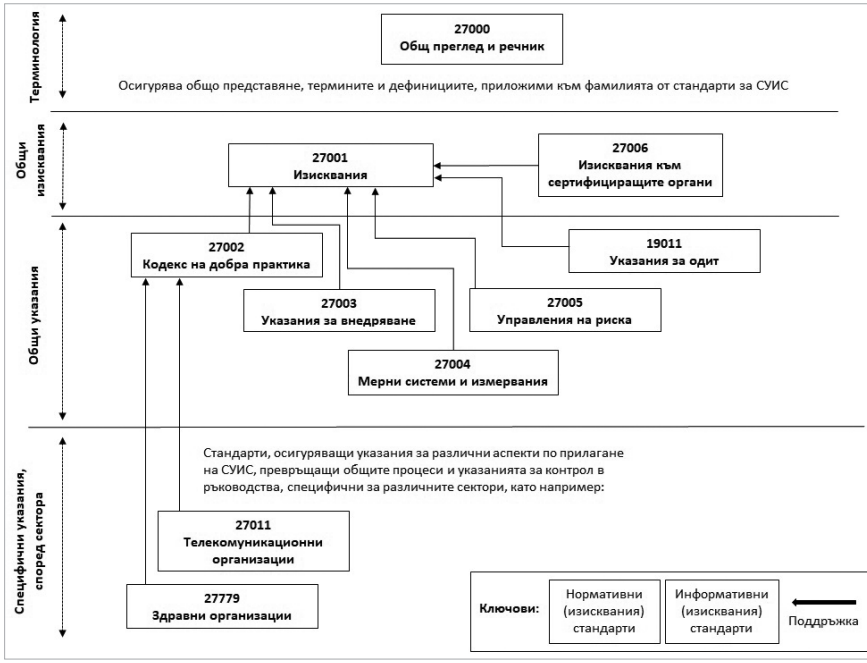
На фигура 2 е показана схема за съотносимостта на стандартите по отношение на сигурност и безопасност на електротехническите системи (включително програмируеми устройства и IoT), съгласно ISO/IEC Ръководство 51.

Фигура 2. Стандарти за сигурност и безопасност

Втора глава
СТАНДАРТИ ЗА ИНФОРМАЦИОННА СИГУРНОСТ

1. Стандартите от серията ISO 27000

В отговор на всички изисквания и нужди на организациите, свързани с управление и защита на информацията, Международната организация по стандартизация (ISO) създава широкообхватна система от стандарти за информационна сигурност, т.нар. ISO/IEC 27000-серия (още позната като „фамилия стандарти за СУИС“, или „ISO27k“, фиг. 3).



Фигура 3. Стандарти от серията ISO 27000

1.1. Стандартът ISO 27001:2017 – Системи за управление на информационната сигурност. Изисквания. Този стандарт е един от малкото задължителни за внедряване и сертификация стандарти

в България. Той е задължителен естествено само за държавната, териториалната и местната администрации, съдебната власт и за лицата осъществяващи публични функции и за организациите предоставящи обществени услуги. Такива са:

- Лица осъществяващи публични функции – са нотариусите, частните съдебни изпълнители, държавните и общинските учебни заведения, държавните и общинските лечебни заведения и други лица и организации, чрез които държавата упражнява своите функции и на които това е възложено със закон.
- организации предоставящи обществени услуги, като образователни, здравни, водоснабдителни, канализационни, топлоснабдителни, електроснабдителни, газоснабдителни, телекомуникационни, пощенски или други подобни услуги, предоставени за задоволяване на обществени потребности, включително като търговска дейност, по повод на чието предоставяне могат да се извършват административни услуги.

Задължителното внедряване и сертификация по този стандарт е обусловено от Наредбата за общите изисквания за оперативна съвместимост и информационна сигурност, издадена на основание чл. 43, ал. 2 от Закона за електронното управление.

Серията предлага система от най-добри практики и препоръки за управление на сигурността на информацията, оценка на рисковете и въвеждане на контроли в контекста на цялостната СУИС, която е проектирана по начин, който я прави съвместима структурно и функционално със системите за управление, разработени и въведени с другите стандарти на ISO, като например за управление и гарантиране на качеството (QMS на серията ISO 9000) или за управление на ИТ (ITMS на серията ISO 20000). Серията ISO27k е създадена съзнателно с максимално широк обхват както по отношение на качествените параметри и техническите условия и въпроси на сигурността на информацията, така и на обхванатите организации по тип, размери и сфера на дейност. ISO/IEC 27000 регламентира **изискванията** за *внедряване, управление, документиране и непрекъснато усъвършенстване на СУИС* с приложение

на подход, базиран на **управление на риска** за сигурността на информацията в организацията чрез установяването му в предварително определен диапазон на допустими стойности. По този начин дейностите по внедряване на системата СУИС се съсредоточават в посока **идентифициране, анализ и оценка на рисковете и намаляването им до предварително зададено приемливо ниво**, като за целта се използва съдържащата се в стандарта система от контролни точки за оценка на риска.

Основа за прилагането на серията от стандарти са т.н. контроли. Контролите представляват списък от добри практики за осигуряване на различните аспекти на информационната сигурност: организационни и технически. Дефинираните контроли регламентират и регулират областите, свързани с: политика за сигурност; персонал; оборудване; контрол на достъпа до компютърни системи и данни; съответствие на законовите изисквания и стандарти; придобиване, развитие и поддържане на СУИС; управление на непрекъснатостта на бизнеса.

1.2. ISO/IEC 27000:2018 Информационни технологии – техники за сигурност – Преглед на СУИС и речник – Представя общ преглед и речник на системите за управление на информационната сигурност, които представляват предмет на семейството стандарти за СУИС и дефинира съответните термини и определения.

1.3. ISO/IEC 27001: 2013 (БДС 2017) – Информационни технологии. Методи за сигурност. Изисквания за системи за управление на информационната сигурност – Определя изискванията за създаване, внедряване, поддържане и непрекъснато подобряване на СУИС в рамките на организацията. Включва също изисквания за оценка и третиране на рисковете за сигурността информацията, в съответствие с потребностите на организацията. Дефинираните изисквания са общи и са предназначени да бъдат приложими за всички организации, независимо от техния вид, размер или характер.

1.4. ISO/IEC 27002: 2013 (БДС 2017) Информационни технологии. Методи за сигурност. Техники за сигурност на СУИС

– допри практики – Дава указания за приложение на практики за управление на сигурността на информацията, включващи избор, прилагане и управлението на контроли, в съответствие с рисковете на средата за информационната сигурност на организацията.

1.5. ISO/IEC 27003: 2017 Информационни технологии. Методи за сигурност. Указания за внедряване на системи за управление на информационната сигурност – Фокусира се върху най-важните аспекти на процесите, свързани с успешното иницииране, планиране и дефиниране на проект за внедряване на СУИС в съответствие с ISO/IEC 27001. Описва процесите от първоначалното получаване на одобрение от ръководството за приложение на СУИС, до създаване на окончателен план за проект на СУИС.

1.6. ISO/IEC 27004:2016 Информационни технологии. Методи за сигурност. Управление на информационната сигурност. Мерни системи и измервания на СУИС – Предлага ръководство за разработване на мерки и за осъществяване на измервания, с цел се оцени ефективността на внедрената СУИС и на контролите, или на групи от контроли, в съответствие със спецификацията в ISO/IEC 27001.

1.7. ISO/IEC 27005:2018 Информационни технологии. Методи за сигурност. Управление на риска при системите за управление на сигурността на информацията – Предлага ръководство, предназначено да подпомогне успешното осигуряване на информационната сигурност, на основа на подход за оценка и управление на риска. Заложените в стандарта етапи, свързани с управлението на риска са:

- Установяване на контекста;
- Идентифициране на риска;
- Анализ на реска;
- Преценяване на риска;
- Третиране на риска.

Като този процес се извършва задължително най-малко веднъж в рамките на всеки одитен цикъл, но може да се наложи и непланиран анализ, свързан с изменения във външната среда, като нормативни актове, осъществена кибератака по отношение на организацията, открити и съобщени от производителите съществени уязвимости на хардуера или софтуера и др.

1.8. ISO/IEC 27006:2015 Информационни технологии. Методи за сигурност. Изисквания за сиртификационни органи на системи за управление на информационната сигурност – Предлага система от изисквания, която трябва да гарантира, че органите, издаващи сертификати за СУИС притежават и демонстрират необходимите компетентност и надеждност. Стандартът съдържа и ръководство, което предлага допълнителни интерпретации на посочените изисквания.

1.9. ISO/IEC 27007:2011 Информационни технологии. Методи за сигурност. Указания за одит на системи за управление на информационната сигурност – Предоставя указания за управление на програми за одит на СУИС и за извършване на одитите. Приложим е за организациите, които искат да провеждат вътрешни или външни одити на СУИС. Заменен е през 2018 година с единен стандарт за извършване на одити за всички сертификати от сериите ISO – ISO 19011:2018 (БДС 2018) Указания за извършване на одит на системи за управление.

Изброените до тук осем стандарта са т.н. „общи“ стандарти от серията ISO27k, отнасящи се за базовите принципи, изисквания, внедряване, процедури за оценка на риска и одит на СУИС, които са адаптирани и приети като национални стандарти от БИС (Български институт по стандартизация). Цялата серия наброява 34 стандарта, намиращи се в различни стадии на подготовка и приложимост, отнасящи се основно за различни частни случаи на внедряване на СУИС и ръководства за приложение в специфични области (напр. управление на междусекторни и междуорганизационни комуникации) и индустрии, като например телекомуникации, финанси, мрежи и др.

Интеграцията на групата стандарти за СУИС и връзката им с останалите стандарти за системи за управление на ISO се осъществява чрез обобщаващия ISO/IEC 27000:2018.

1.10. ISO/IEC 27000:2018 Общ преглед и речник, който съдържа:

- Общ преглед на серията стандарти ISO27k, който показва как те се използват съвместно за планиране, внедряване, сертифициране и експлоатиране на СУИС, от гледна точка на осигуряване на информационна сигурност и управление на риска;
- Речник, съдържащ акуратно формулирани официални дефиниции на термините, свързани с информационната сигурност, използвани в ISO27k.

Групата базови, общо приложими части на серията стандарти ISO27k са свързани в логическа последователност помежду си, разглеждат се като единна структура и се прилагат комплексно, тъй като регламентират фундаменталните изисквания за реализация на СУИС от организациите:

- ISO 27001 формулира обхвата и изискванията, накоито трябва да отговаря СУИС;
- ISO 27002 осигурява методика, на база обобщение на добрите практики, за определяне на контроли и последователност от дейности, които трябва да осъществи организацията за да постигне дефинираните в първата част изисквания;
- ISO 27003 предлага на организациите указания за планиране, разработване и прилагане на проект за внедряване на СУИС, в съответствие с изискванията на ISO 27001;
- и накрая, чрез предложената в ISO 27004 система за измерване се прави оценка на постигнатите резултати по внедряване на СУИС.

Оценката и сертифицирането на организациите се извършва по регламенти и от органи, функциониращи на база указанията и препоръките на ISO 27006 и ISO 190011.

2. ISO/IEC 31000, Управление на риска – Принципи и насоки (ISO/IEC 31000, Risk management – Principles and guidelines) и ISO/IEC 31010:2019, Управление на риска – Техники за оценка на риска (ISO/IEC 31010, Risk management – Risk assessment techniques)

IEC 31010:2019 е публикуван от БДС като стандарт с двойно лого с ISO и предоставя указания за избора и прилагането на методи за оценяване на риска в най-различни ситуации. Методите се използват за оказване на съдействие при вземането на решения, когато има неопределеност, за предоставяне на информация за конкретни рискове и като част от процеса за управление на риска. Документът съдържа обобщения на редица методи, като са позовани, и други документи, в които методите са описани по-подробно. Това второ издание отменя и заменя първото издание, публикувано през 2009 г. Това издание представлява технически преработено издание. Това издание включва следните значителни технически промени по отношение на предишното издание:

- дадени са повече подробности за процеса на планиране, прилагане, проверка и валидиране на използването на методите;
- увеличен е броят и обхватът на приложение на методите (Български институт за стандартизация https://bds-bg.org/bg/project/show/bds:proj:105683).

3. Специална публикация (SP – Special Publication) 800-37 ревизия 2 на Националния институт за стандарти и технологии на САЩ (NIST) е рамка за управление на риска за информационни системи и организации: подход за жизнения цикъл на системата за сигурност и поверителност.

NIST SP 800-37 rev 2 е публикуван през декември 2018 г. и описва рамката за управление на риска (RMF) и насоки как да се прилага RMF към информационните системи. Специалната публикация е в съответствие с изискванията на Службата за управление и бюджет (OMB), по-специално с циркуляра а-130 на OMB. RMF очертава необходимата структура и процеси за управление на сигурността, поверителността и риска. Рамката включва информация за категоризацията на сигурността, която управлява,

прилагане, оценка и непрекъснато наблюдение. Целта на RMF е да подготви организациите да изпълняват подходящи дейности за управление на риска през жизнения цикъл. Рамката също така предоставя пътна карта за киберсигурност, за да осигури управление на риска в почти реално време на информационните системи с дърво на решенията, поддържащо поверителността и сигурността. Роли и отговорности и Резюме на RMF задачите също могат да бъдат намерени в NIST SP 800-37 ревизия 2 за установяване на отчетност и отговорност за контролите в рамките на информационните системи на организацията.

4. ISO 22301:2019 (БДС EN ISO 22301:2020) Сигурност на обществото. Системи за управление на непрекъснатостта на дейността. Изисквания.

Нарастващото използване на ИКТ услуги поставя фирмите в сериозна зависимост от доставчика. Цифровата зависимост се дефинира като критична зависимост на изпълнението на основните функции и дейности на институции, организации, бизнеси и обществото като цяло от ИКТ.

От своя страна устойчивостта (Resilience, NIST) е способност, свойство (на организацията) бързо да се адаптира и да се възстановява от известни или неизвестни промени в околната среда чрез цялостно и последователно реализиране на управлението на риска, управление при извънредни ситуации и планиране на непрекъснатост на дейностите/операциите.

Международният стандарт ISO 22301 определя изискванията за създаване и управление на ефикасна система за управление на непрекъснатостта на дейността (СУНД). ISO 22301 описва основните елементи на управлението на непрекъснатостта на бизнеса: планиране, разработване, внедряване, функциониране, мониторинг, преглед, анализ и непрекъснато усъвършенстване. Изискванията на стандарта са предназначени да допринесат за: непрекъснатостта на дейността на организацията, за защита и намаляване вероятността от поява на инциденти, подготовка за реакция и възстановяване при възникване на разрушителни инциденти.

Системата за управление на непрекъснатостта на дейността подчертава важността на:

- разбирането на потребностите на организацията и на необходимостта от установяване на политика и цели за управление на непрекъснатостта на дейността;
- внедряването и прилагането на механизми за контрол и мерки за управление на цялостната способност на организацията да управлява непрекъснатостта при разрушителни инциденти;
- наблюдение и проверката ефикасността на СУНД;
- непрекъснатото подобряване на базата на обективни измервания.

СУНД като всяка друга система за управление включва следните основни елементи: политика, персонал с определени отговорности, процеси за управление, документация, осигуряваща достоверни доказателства за одит, всички процеси за управление на непрекъснатостта на дейността, приложими към организацията.

Международният стандарт ISO 22301 прилага модела „Plan-Do-Check-Act“, характерен за повечето ISO стандарти. Този модел позволява СУНД да бъде интегрирана с други стандарти за системи за управление, например с ISO 9001 „Системи за управление на качеството“, ISO 14001 „Системи за управление по отношение на околната среда“, ISO/IEC 27001 „Системи за управление на сигурността на информацията“, ISO/IEC 20000-1 „Информационни технологии. Управление на услугите“ и ISO 28000 „Спецификация за системи за управление за сигурност на веригата за доставки“, като по този начин спомага за ефикасно функциониране на процесите в организацията и тези по одитиране на системите за управление, внедрени от нея.

Изискванията, определени в ISO 22301, са общи и са предназначени да се прилагат от различни организации, или части от тях, независимо от вида, големината и естеството на работа на организацията. Степента на прилагане на тези изисквания зависи от конкретните условия и контекст на организацията.

Прилагането на стандарт ISO 22301 в управлението на непрекъснатостта на бизнеса е от особено значение за организации,

които работят в условия на висока степен на риск, като производствени предприятия с клонове, разположени в различни региони, в публичния сектор, финансите, транспорта, телекомуникациите, където способността за осигуряване на непрекъснатостта на дейността е от съществено значение както за самата организация, така и за нейните клиенти, заинтересовани страни и обществото като цяло.

Внедряването и сертифицирането съгласно ISO 22301 на организацията би довело до редица предимства:

- максимално бързо възстановяване на нормалната работа на организацията в случай на непреодолима сила или кризисна ситуация, включително кибератака;
- готовност за реакция при кризи, основана на надежден и представителен анализ на рисковете;
- постоянна ангажираност и готовност на персонала за работа в случаи на форсмажорни ситуации;
- чрез внедряване и сертификация на СУНД се гарантира доверието в способността на организацията да поддържа непрекъснатостта на дейността и да увеличи доверието у потребителите, правителствени, неправителствени и други заинтересовани страни от дейността.

5. Система за управление на информационната сигурност съгласно стандартите от серията ISO 27k

Системата за управление на информационната сигурност съгласно ISO 27001:2017 (СУИС), се базира освен на него и на приложимите нормативни изисквания. Стандарта е приложим във всяка организация (не само посочените по-горе) и е напълно интегрируем с изискванията на ISO 9001:2015 и ISO 14001:2015.

Системата за управление на информационната сигурност обхваща:

- **Оценка на риска за информационната сигурност** – включващ идентификация на активите имащи отношение към сигурността, определяне на заплахите към тях, оценка, третиране и управление на рисковете;

● **Сигурност на човешките ресурси** – регламентираща опазването на поверителността на информацията от персонала;

● **Физическа сигурност** – включваща изисквания за прилагане на средства за физическа защита на активите, както и за гарантирането на сигурността им от заплаши на околната среда – пожари, наводнения, извънредни ситуации;

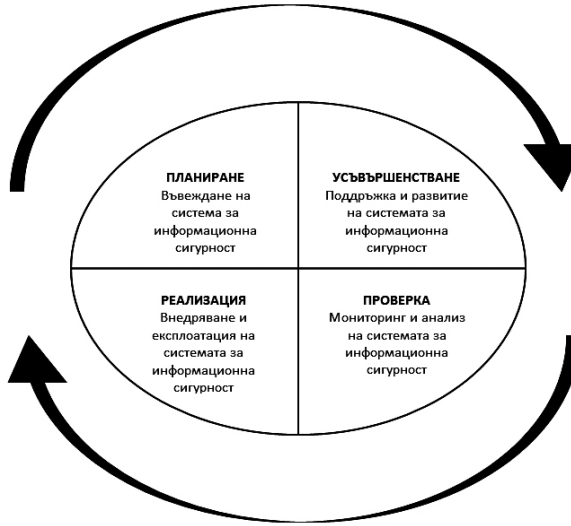
● **Компютърна и мрежова сигурност** – регламентирана в няколко раздела на приложение А на стандарта. Включва изисквания за архивиране, защита от вируси и външни атаки, организация на логическия достъп до системите и мрежите, контрола на мрежовите услуги и други;

● **Сигурност при разработването на софтуер и хардуер** – тази част включва изисквания за въвеждане на механизми за контрол на сигурността на софтуера и хардуера още на етап на неговото разработване;

● **Управление на инциденти** – включва изисквания за докладване, реакция и анализ на инцидентите по сигурността;

● **Управление на непрекъсваемостта** – този процес създава рамка за въвеждане на непрекъснат режим на работа на информационните активи, като по този начин се гарантира постоянното и качествено обслужване на клиентите.

Системата за управление на информационната сигурност представлява модел за създаване, внедряване, експлоатация, мониторинг, преглед, поддържане и подобряване на защитата на информационните активи на организацията, за да се осигури постигане на нейните бизнес цели, базирани на оценка на риска спрямо дефинирани нива на допустимия риск, определени от гледна точка на ефективно третиране и управление на риска. За успешното внедряване на СУИС е необходимо да бъде осъществен и поддържан следния процесен цикъл: анализ на изисквания за сигурност на информационните активи и прилагане на подходящи контроли и проверки, които да гарантират тяхната защита, измерване и анализ на резултатите от действието на системата, коригиращи мерки и дейности за подобряване на системата. Тези дейности се извършват в рамките на всеки одитен цикъл (фиг. 4).



Фигура 4. Одитен цикъл на СУИС

Базовите стандарти от серията ISO27k, които предлагат модел за проектиране, внедряване, експлоатация и поддържане на СУИС са ISO27001 и ISO27002. Като интегрирана съставна част на стандартите на ISO по създаване и внедряване на различни системи за управление, серията ISO27k възприема процесния подход за осигуряване на реализацията на СУИС от организациите. С понятието „процесен подход“ се означава съвместното прилагане на система от процеси в организацията – идентифицирането и взаимодействието на тези процеси и тяхното управление. Подобно на другите стандарти на ISO, използващи процесния подход, серията ISO 27k възприема и прилага процесния модел PDCA („Plan-Do-Check-Act“) за структуриране на включените в СУИС процеси. Концепцията включва непрекъсната обратна връзка и подобряване на дейностите, което е базов принцип на подхода PDCA, като стремежът е да се следи динамиката и промените на заплахите и уязвимостите и въздействието на инцидентите върху сигурността на информацията. Отделните фази на модела PDCA са:

1. Планиране (Plan) – въвеждане на СУИС: Създаване политика, цели, процеси и процедури за СУИС, отнасящи се до упра-

вление на риска и подобряване на сигурността на информацията за постигане на резултати в съответствие с общите политики и цели на организацията.

2. Реализация (Do) – внедряване и експлоатация на СУИС: Внедряване и задействане на СУИС политиката, контролите, процесите и процедурите.

3. Проверка (Check) – мониторинг и анализ на СУИС: Извършване на оценка, и където е приложимо, измерване на реализацията на процесите в съответствие със СУИС политиката и целите, докладване на резултатите на ръководството за анализ.

4. Усъвършенстване (Act) – поддръжка и развитие на СУИС: Предприемане на превантивни и корективни и действия, базирани на резултатите от вътрешния одит на СУИС, анализ на управлението и друга значима информация, за да се постигне непрекъснато подобрение на СУИС.

Като част от СУИС, рисковете асоциирани със информационните активи на организацията трябва да бъдат адресирани. Постигането на информационна сигурност изисква да бъде управляван риска, който обхваща видовете риск, асоциирани с физически, човешки и технологични заплахи, отнасящи се за всички форми на информация, използвани от организацията. Внедряването на СУИС е стратегическо решение за организацията, затова е необходимо тя да бъде старателно проектирана, интегрирана и актуализирана в съответствие с нуждите на организацията, т.е. не може да се направи машинално прехвърляне на дадена система от една организация в друга. Тя трябва да е съобразена с целите на организацията, изискванията за сигурност, използваните бизнес процеси и от размера и структурата на организацията. Процеса на проектиране трябва да бъдат отразени интересите и изискванията по сигурността на информацията на всички заинтересовани страни на организацията, включително клиенти, доставчици, бизнес партньори, акционери и трети страни. Организациите и техните информационни системи и мрежи са изправени пред заплахи от широк спектър от източници на заплахи, включващи компютърни измами, шпионаж, саботаж, вандализъм, както и от различни физически заплахи и природни бедствия.

Тъй като различните видове атаки са подробно класифицирани и описани в различни източници те няма отново да бъдат описани тук. Информация за тях може да бъде открита в Кръстев, 2021, Boldea, 2019 и др., както и в специализираните библиотеки за уязвимости.

Взаимодействието между частните и публичните информационни мрежи, мащаба на обмена и споделянето на информационните активи увеличава трудността да се контролира достъпа до, и обработката на информацията. В допълнение, разпространението на мобилни устройства за обработка и съхранение на информационни активи може да отслаби ефективността на традиционните контроли.

Внедряването на изискванията на стандартите позволява на организациите да управляват сигурността на своите информационни активи и да извършат подготовка за осъществяване на независима оценка за възможностите на тяхната СУИС да осигурява защита на информацията под всякаква форма – финансова информация, интелектуална собственост, данни за персонала и лични данни, информация, поверена им от техните потребители или от партньори и все повече има характер на задължително условие, вместо на стратегическо предимство, както се считаше по-рано. Информацията може да бъде съхранявана под множество форми, включително дигитална форма (например файлове с данни, съхранявани на електронни или оптични носители), материална форма (например на хартия), както и нерегистрирана информация под формата на знания на служителите. Информацията може да бъде предавана по различни начини, включващи веществено пренасяне от куриери, електронна или вербална комуникация. Независимо под каква и форма се намира, или средствата, чрез които се предава, тя винаги се нуждае подходяща защита. Системите за управление на сигурността на информацията, изградени в съответствие със стандарта ISO 27000 третират всички видове и форми на информацията, но за целите на това изследване ще разгледаме само информацията, генерирана или съхранявана и обработвана в цифров вид от информационните системи на организацията.

Информацията на организацията във все по-голяма степен се свързва и зависи от информационните и комуникационните технологии. Тези технологии са съществен елемент във всяка организация

като подпомагат и улесняват създаването, обработването, съхраняването, предаването, защитата и унищожаването на информацията. Информационната сигурност изисква прилагането и управлението на подходящи мерки за сигурност, което включва разглеждането и отчитането на широк кръг от заплахи, с цел да се гарантира непрекъснатост на бизнеса и траен успех, както и за намаляване до минимум на потенциалните вредни въздействия. Постига се чрез използване на приложна система от контроли, селектирани в съответствие с избрания процеса на управление на риска и се управлява с помощта на СУИС, включваща политики, процеси, процедури, организационни структури, софтуер и хардуер за защита на идентифицираните информационни активи. Контролите, необходими за гарантиране на ИС трябва да са избрани по начин, който да позволява безпроблемното им интегриране с бизнес процесите на организацията и системата за управление на качеството ако такава е внедрена и сертифицирана в организацията, доколкото често организациите сертифицират едновременно ISO 27000 и ISO 9000 – Системи за управление на качеството или ISO 20000 – Системи за управление на услугите.

6. Изисквания към СУИС

Стандартът ISO 27001 предлага списък на базовите, общи изисквания, които е необходимо да се удовлетворят за да реализира и притежава дадена организация СУИС и за да се сертифицира в областта на информационната сигурност. Всяка организация трябва да създаде, внедри, експлоатира, контролира, преразглежда, поддържа и усъвършенства създадената СУИС в съответствие с дейността на организацията и риска, пред който се изправя, като с цел реализацията на тези процеси се използва предлагания от ISO 27001 модел PDCA. Източниците на изисквания по отношение на сигурността са три:

– оценка на рисковете за организацията, като се вземат предвид бизнес стратегията и целите. Чрез оценка на риска се идентифицират заплахите за активите и вероятността за появяване на уязвимости, както и потенциалното им въздействие върху организацията. Организацията трябва да дефинира нивото на т.н. приемлив риск (риск, който ако се случи ще доведе по последствия, които организацията може да понесе без значително разстройство на дейността си)

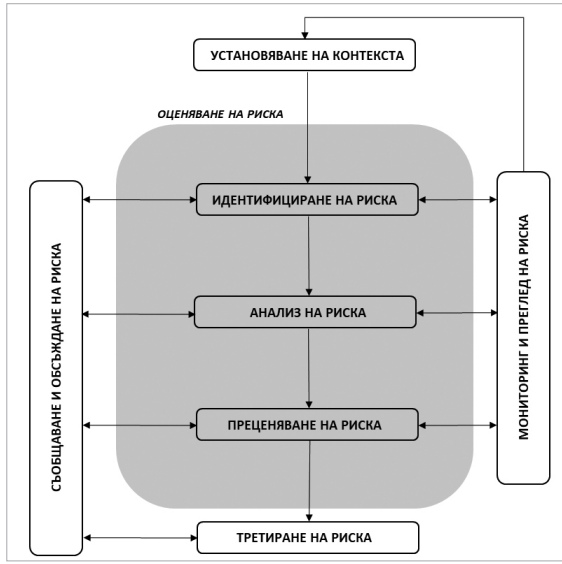
и на тази основа да се определят действията за предотвратяване (или минимизиране на ефекта) на недопустимите рискове и довеждането им до допустими за организацията граници;

– правните, законови, регулаторни и договорни изисквания, на които организацията, нейните търговски партньори, контрагенти и доставчици на услуги трябва да отговарят;

– конкретният набор от принципи, цели и бизнес изисквания за обработка на информацията, които организация развива за да осъществява своите операции.

Няма да разглеждаме подробно процесите по въвеждане и поддържане на СУИС, но ще обърнем внимание на тези от тях, които имат отношение към техническите аспекти на осигуряване на сигурността на информацията, като не забравяме, че тези процеси са свързани със съответните организационни аспекти по разработване на политики, правила, процедури и други мерки, които са неизменна част от процеса и основа на неговото документиране.

Един от съществените елементи, от които до голяма степен зависи по-нататъшната ефективност на системата е оценката на риска (фиг. 5).



Фигура 5. Процес на управление на риска

Това е и една от най-чувствителните фази на процеса, провала на която води често до провал на цялостното въвеждане на СУИС в организацията. За реализацията на този процес, организацията трябва да извърши следните стъпки:

– дефиниране на обхвата и границите на управлението на информационната сигурност в зависимост от условията и характеристиките на бизнеса, както и от самата организация, нейното местоположение, активи, технологии, като се включват и изключения от дефинирания обхват, които трябва да бъдат добре обосновани;

– дефиниране на политиката – стратегическите цели по отношение на сигурността. На тази стъпка от процеса трябва да се идентифицират зависимостите на сигурността от околната среда (нормативна уредба, партньори, контрагенти и свързаните с тях изисквания по отношение на сигурността), дефинира и политиката по отношение на рисковете (критерии за оценка на риска, ниво на приемливия риск);

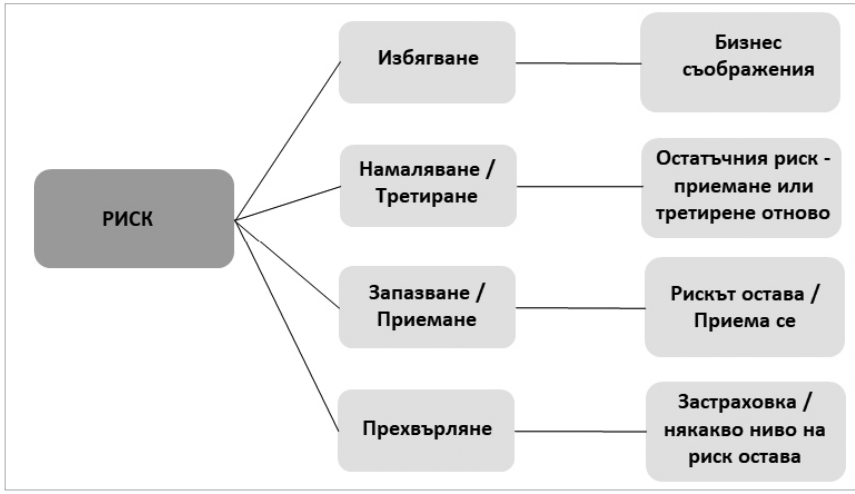
– дефиниране на методологията за оценка на риска в организацията, критериите за допустим и приемлив риск;

– идентифициране на рисковете: определяне на активите в рамките на СУИС и собствениците на тези активи, идентифициране на заплахите за активите, идентифициране на уязвимостите, които може да бъдат използвани от заплахите, определяне на въздействието върху активите, което може да окаже загубата на конфиденциалност, цялостност и достъпност;

– анализ и оценка на риска: оценка на въздействието върху бизнеса на организацията, което може да се получи в резултат на пробив в сигурността, от гледна точка на последствията от загубата на конфиденциалност, цялостност и достъпност на активите, оценка на вероятността да се случи такъв пробив в сигурността, в предвид преобладаващите заплахи и уязвимости и съществуващите вече контроли, както и възможните въздействия върху активите, оценка на нивата на риска, определяне дали рисковете са приемливи или се изисква тяхното третиране, като се използват установените критерии за приемливия риск;

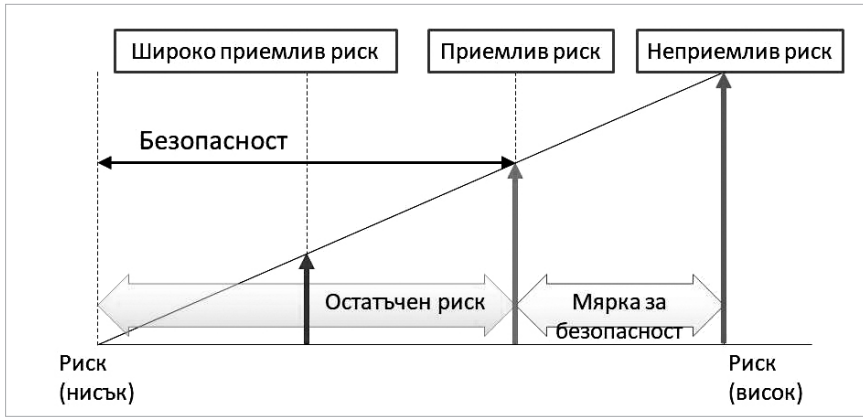
– идентифициране и оценяване на възможностите за обработване на рискове (фиг. 6). Възможните действия включват: при-

лагане на подходящи контроли, съзнателно и целенасочено приемане на рисковете, при условие, че те ясно удовлетворяват политиката на организацията и критериите за допустимост на риска, отхвърляне на риска, прехвърляне на бизнес риска чрез застраховане или аутсорсинг (при което съществува и остатъчен риск за организацията и важното е той да е поносим);



Фигура 6. Варианти за третиране на риска

– селектиране на контролни цели и контроли за обработване на рискове. Контролните цели и контроли трябва да бъдат избрани и реализирани така, че да удовлетворяват изискванията на процесите за оценка на риска и за обработка на риска. Този избор трябва да отговаря на критерия за приемане на рискове, както и на правните, регулаторни и договорни изисквания.



Фигура 7. Видове риск спрямо неговото ниво и нивото на приемливия риск

Безопасността на практика означава, че рисковете, които поемаме са допустими (достатъчно малки) за организацията. Приет риск при дадени обстоятелства зависи от ценностите на обществото, мисията и значението на организацията. Трябва да не забравяме, че остатъчен риск съществува дори в рамките на безопасността и той не може да бъде елиминиран а само да бъде намалено нивото му.

На основата на анализа се разработва и плана за третиране на риска, който е основа за оценяване и подобряване на нивото на сигурността. Планът дефинира подходящи действия, ресурси, отговорности и приоритети за управление на риска по отношение на информационната сигурност.

Този процес не е еднократен и се извършва най-малко веднъж в рамките на одитния период, но може да бъде предизвикан и от събития, като съществен пробив в сигурността, открити и докладвани сериозни уязвимости или начини за тяхното използване, развитие на технологиите, промяна в законодателството и други съществени изменения в околната среда или в самата организация. Периодичния мониторинг има за цел оценка на ефективността на системата и нейното непрекъснато усъвършенстване в съответствие с новите предизвикателства.

Документирането на СУИС е една от важните организационни задачи, свързани с разработването на съответните документи и поддържането им в актуално състояние, за което могат да бъдат използвани системи за управление на документите. За целите на това изследване обаче ще се фокусираме върху документирането на оперативната работа по прилагането на СУИС, което се извършва основно чрез записите. Записите са и един от основните източници на доказателства и в процесите на оценка и одит. По определение записът е документ, съдържащ получени резултати или предоставящ доказателства за извършени дейности (ISO/IEC 27000:2018). Записите могат да бъдат и в електронен вид, т.е. като записи по сигурността могат да бъдат разглеждани лот файлове, електронни съобщения, извлечения от електронни регистри, извлечения от системи за мониторинг и други.

7. Контрол на записите

Записите трябва да бъдат създадени и поддържани, за да се осигури доказателство за съответствие с изискванията и за ефективността на функциониране на СУИС. Те трябва да са защитени и контролирани, трябва да бъдат четливи, лесно разпознаваеми и възстановими. Необходимо е установяването на контроли за идентификация, съхраняване, защита, изтегляне, времето за запазване и на унищожаване на записите и те трябва да бъдат документирани и приложени. Записите трябва да съхраняват изпълнението на процесите и всички случаи на значителни инциденти със сигурността, свързани с СУИС.

8. Контроли по сигурността

Контролите по сигурността са на практика мерките, които организацията предприема, за да контролира процесите по управление на сигурността. Избора и прилагането им е пряк резултат от оценката и анализа на риска. На 15.02.2022 г. международната организация ISO публикува новата версия на ISO/IEC 27002:2022 „Information security, cybersecurity and privacy protection — Information security controls“.

Стандартът за управление на информационната сигурност ISO/IEC 27001 „Информационни технологии. Методи за си-

гурност. Системи за управление на сигурността на информацията. Изисквания“ и неговият Кодекс за добра практика (**ISO 27002**) са последно актуализирани преди почти 10 години. Очаква се и ревизирана версия на ISO/IEC 27001.

Първото, което веднага се забелязва, е промяната в заглавието – „Кодекс за добра практика“ е заменено с „Контроли на информационната сигурност“, което по-добре отразява целта на стандарта – да предостави указания за прилагането на референтен набор от контроли за информационна сигурност. Стандартът е значително по-дълъг от предишната версия, а самите контроли са прегрупирани и актуализирани. Промените целят опростяване на внедряването: броят на контролите е намален от 114 на 93 (чрез обединяване), разпределени са в 4 групи, вместо в предишните 14. Има 11 нови контроли, като нито една от досегашните не е премахната.

Новият ISO/IEC 27002:2022 има 93 контроли, разпределени в 4 групи, както и две приложения:

- Организационни (клауза 5);
- Свързани с човешките ресурси (клауза 6);
- Свързани с физическата сигурност (клауза 7);
- Технологични контроли (клауза 8);
- Приложение А – Използване на атрибути;
- Приложение В – Съответствие с ISO/IEC 27002:2013.

Новите контроли, добавени към стандарта са:

- Разузнаване за заплахи;
- Информационна сигурност при използване на облачни услуги;
- ICT готовност за непрекъснатост на бизнеса;
- Мониторинг на физическата сигурност;
- Управление на конфигурацията;
- Изтриване на информация;
- Маскиране на данни;
- Предотвратяване на изтичане на данни;
- Дейности по наблюдение;
- Уеб филтриране;
- Сигурно писане на код.

Контролите вече могат да бъдат определени и групирани чрез пет типа атрибути, за да се категоризират по-лесно:

- Тип контрол (превантивен, разпознаващ, коригиращ);
- Свойства на информационната сигурност (поверителност, цялост, наличност);
- Концепции за киберсигурност (идентифициране, защита, откриване, реагиране, възстановяване);
- Оперативни възможности (ръководни дейности, управление на активи и др.);
- Домейни на сигурност (управление и екосистема, защита, отбрана, устойчивост).

Очакванията са, че ISO няма да публикува изцяло нова версия на стандарта ISO 27001 а че ще има изменение, наречено ISO/IEC 27001:2013/DAMD 1, в което Приложение А ще бъде заменено с нормативна версия на 93-те нови контроли от ISO 27002:2022.

9. Разлика между сигурност и поверителност на данните

Сигурността е свързана със защитата на данните от всеки вид, докато поверителността е свързана със защитата на самоличността на потребителя и неговите лични данни. Конкретните разлики обаче, са по-сложни и със сигурност може да има области на припокриване между двете.

Сигурността се отнася до защита срещу неоторизиран достъп до данни. Въвеждането на контроли за сигурност може да ограничи достъпа до информацията. Поверителността е по-скоро свързана със защита на личните данни на физическите лица и тяхното неразпространение извън границите на необходимостта. Ако информационните системи и данните, които съдържат, са били компрометирани, поради недостатъчна сигурност, произтичащата от това загуба на данни, може да има значими последици за лицата, чиито данни се съхраняват в тези системи.

Най-лесно разликата между двете понятия може да се илюстрира с пример. Ако персоналът на болница или поликлиника използва защитени системи, за да комуникира с пациентите, относно тяхното здраве, вместо да изпраща информация чрез лични имейл

акаунти, то този тип предаване на данни е пример за сигурност. От друга страна, разпоредбите за поверителност могат да ограничат достъпа до здравните досиета на пациентите до конкретни членове на болничния персонал, като лекари, медицински сестри и медицински асистенти. Поверителността може също така да определя кога потребителите имат достъп до конкретна информация (т.е. само работни часове), или само до ограничен кръг от пациенти (от конкретно отделение, например), или ограничена част от наличната информация (например само до предписаните на пациента лекарства).

Скорошно проучване (Khag, 2017) се фокусира върху практиките за сигурност и поверителност на над 300 амбулаторни клиники за ХИВ във Виетнам. Проучването установява, че повечето служители са имали подходящи мерки и практики за поддържане на сигурността на данните; въпреки това, защитата на поверителността на пациентите, особено за достъп до данни, споделяне и трансфер, все още изисква подобрение.

Въпреки че концепциите за сигурност и поверителност са заплетени, е ясно, че е възможно да имаме сигурност без поверителност, но е невъзможно да имаме поверителност без сигурност.

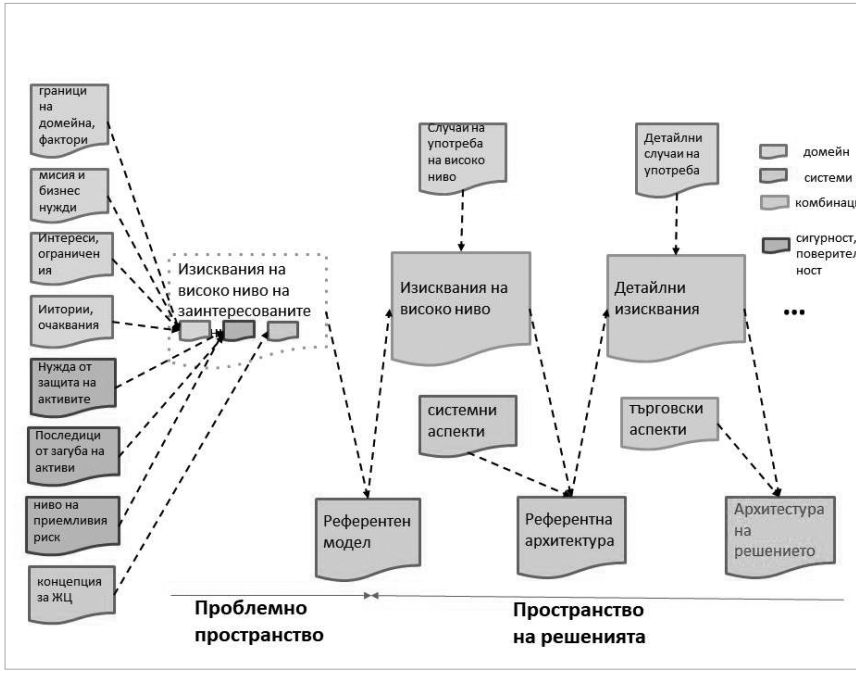
В последната си публикация [1] Ан Кавукиан, комисар по въпросите на поверителността на канадската провинция Онтарио и автор на „Privacy by design“ [2] пише, че въпреки популярността си в областта на киберсигурността, те все още са предизвикателство за бизнеса (Agrawal, 2020), (Bu, 2020). Принципите се заключават в:

1) проактивен, а не реактивен, който се фокусира върху предотвратяване;
2) поверителност като настройка по подразбиране;
3) поверителност, вградена в дизайна;
4) пълна функционалност;
5) сигурност от край до край;
6) видимост и прозрачност;
7) зачитане на поверителността на потребителите.

Трета глава
СИГУРНОСТ ПО ДИЗАЙН

1. Сигурни архитектури

Както вече отбелязахме, системата за управление на сигурността е базирана на процеса на оценка на риска и мерките за неговото управление. Затова е важно да разгледаме по-подробно оценката на риска като система от гледна точка на софтуерните архитекти и ефекта върху тяхната работа. Фигура 8 представя процеса на разработка на архитектурата на едно софтуерно решение, като лявата страна представлява проблемното пространство, т.е. изискванията, които разработваното софтуерно решение се очаква да удовлетвори а в дясната половина е разработеното решение, като то е представено с две нива на детайлност: архитектура на високо ниво и детайлна архитектура.



Фигура 8. Разработка на софтуерна архитектура

На входовете на системата са всички изисквания и гледни точки, свързани с нейната разработка и бъдеща експлоатация и нуждите, които се очаква тя да удовлетвори. От гледна точка на заинтересованите страни това са: различните видове активи; мисия или бизнес нужди, цели за сигурност; концепция на операциите (бизнес процесите); закони, разпоредби и политики, които трябва да бъдат спазени, организационни ограничения. От системна гледна точка това са: системни активи (налични и използвани досега в организацията технологични решения); Възможни решения за архитектура, дизайн и изпълнение; система за самозащита; сигурно управление на системата. И от бизнес гледна точка, това са принципите на софтуерното инженерство, сделки, свързани с третиране на риска, времеви и финансови ограничения.

На изхода като част от резултатите на софтуерното проектиране получаваме: изисквания за сигурност (функционални и нефункционални), правила за верификация и тестване. Анализът на политиката за сигурност, нейните цели и организационните планове за управление на сигурността трябва да доведат до разработване на политика за сигурност на системно ниво.

Заплахите и уязвимостите са универсални и важат за всички потребители на даден софтуер или услуга. Съществуват регистри за публично известни уязвимости в информационната сигурност и експозиции. Тези регистри трябва редовно да бъдат следени, както и публикуваните решения на откритите уязвимости, за да може да се реагира на тях по най-добрия известен начин. Механизмите за противодействие трябва да се заложат още в процеса на проектиране (Al-Matou, 2020). Освен по отношение на софтуера трябва да се внимава и с избора на криптиращи алгоритми, доколкото използването на остарял криптиращ алгоритъм би довел до необосновано високо ниво на сигурност по отношение на поверителността на данните.

Най-популярната библиотека е CWE. Common Weakness Enumeration (CWE™) е списък, разработен от общността, на често срещани типове слабости на софтуера и хардуера, които имат разклонения за сигурността. „Слабости“ са недостатъци, дефекти, бъгове или други грешки в внедряването на софтуера или хардуера,

кода, дизайна или архитектурата, които, ако не бъдат адресирани, могат да доведат до уязвимост на системи, мрежи или хардуер за атака. Списъкът на CWE и свързаната класификационна таксономия служат като език, който може да се използва за идентифициране и описание на тези слабости по отношение на CWE. https://cwe.mitre.org/data/definitions/327.html

CISA – Агенцията по киберсигурност и инфраструктурна сигурност на САЩ поддържа собствена библиотека на уязвимостите, която е достъпна на адрес https://www.cisa.gov/known-exploited-vulnerabilities-catalog

Базата данни за уязвимости, поддържана от Националния институт по стандартизация и технологии на САЩ (NIST – National institute of Standards and Rechnology) е на адрес https://nvd.nist.gov/ Тя дава и оценка на риска (по десетобална скала), който съответната уязвимост може да донесе по отношение на организацията, като най-проблемните уязвимости се оценяват като критични и изискват незабавни действия.

https://vuldb.com е друга библиотека за уязвимости, която поддържа доста добър по отношение на потребителите интерфейс за търсене по доставчик на софтуера, продукт и тип.

Списъкът CWE-327: Use of a Broken or Risky Cryptographic Algorithm (Използвани „счупени“ и рискови криптиращи алгоритми) предоставя информация за алгоритми, за които вече съществува информация, че са разбити и не е желателно да бъдат използвани.

Списък на препоръчителни за използване криптиращи алгоритми може да бъде намерен на страницата BliueKrypt https://www.keylength.com/en/8/

NIST – ресурсният център по киберсигурност на САЩ поддържа собствена програма за валидиране на криптографските алгоритми и информация за резултатите от нея може да бъде намерена на този адрес https://csrc.nist.gov/Projects/cryptographic-algorithm-validation-program

Нивото на неблагоприятно въздействие от атака зависи и от архитектурата на системата, която представлява интерес и доколко-

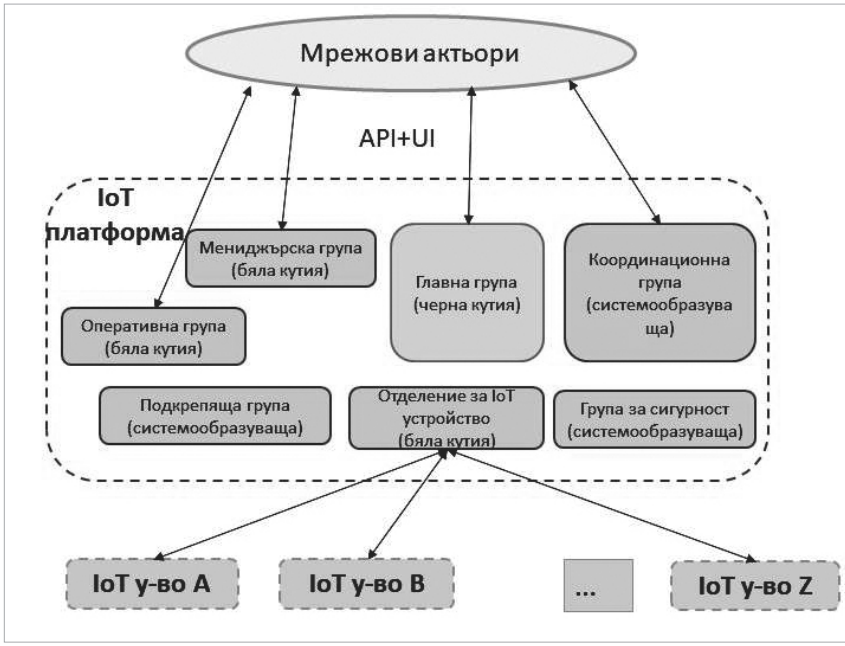
ко тя позволява реализирането на даден вид атака. Сигурността и рискът могат да бъдат обективно свързани чрез архитектура (Kǿien, 2020). Целта е да се използва цялата налична информация, за да се проектират системи със сигурност, безопасност и поверителност по проект и по подразбиране. Архитектът трябва да знае всички връзки между всички артефакти (технически активи, услуги, процеси и т.н.), за да оцени статистически рисковете. Ако внедряването на една система се основава на бизнес процеси, тогава тя може динамично да оценява рисковете. Познавайки нивото на риск, човек може да приложи набор от промени (и/или контроли), за да намали това ниво до приемливо. Всеки системен елемент (материални активи, нематериални активи, хора) трябва да бъде изрично защитен за неговата поверителност, цялостност и достъпност в покой, в транзит и при употреба, през целия си жизнен цикъл (в рамките на жизнения цикъл на системата от интереси).

Връзките между системните елементи се използват, за да се разбере как промените в един системен елемент влияят на други системни елементи и тези взаимоотношения също трябва да бъдат защитени, като в идеалния случай тези връзки са изрични и машинно изпълними.

Системният подход към сигурността разглежда три групи елементи на системата:

- някои системни елементи се третират като черни кутии, като за тях се определя необходимата функционалност, интерфейси, производителност, гаранция за сигурност и т.н.;
- други системни елементи се третират като сиви кутии, като се дефинира и тяхната вътрешна структура (напр. като илюстративни процеси);
- трети системни елементи (които действат като системообразуващи) се третират като бели кутии чрез дефиниране на тяхната (референтна) реализация.

Като пример ще разгледаме архитектура на платформа, обединяваща и управляваща колекция от IoT устройства, представена на фигура 9.



Фигура 9. Архитектура на IoT платформа

Системата се състои от:

– Отделение за IoT устройство за свързване на платформата и различни IoT устройства;

– Поддържаща група за предоставяне на функционалност, споделена в цифрова система (напр. регистриране, наблюдение, обработка на данни, сътрудничество, управление на процеси, управление на решения, анализи и др.);

– Група за сигурност за осигуряване на функционалност за сигурност;

– Основна група за осигуряване на основна бизнес функционалност;

– Координационна група за изпълнение на цифрови договори между различни мрежови участници, IoT устройства и самата платформа;

– Мениджърска група за преконфигуриране на платформата;

– Оперативна група за поддържане на изправното функциониране на платформата.

Доколкото по отношение, както на техническите, така и на софтуерните аспекти на IoT системите множеството от стандарти е в процес на разработка, темата за тяхното сигурно проектиране и изграждане е доста обсъждана и е предмет на редица публикации (Barbosa, 2017), (Nomikos, 2020), (Siavvas, 2022).

Всички управленски и оперативни дейности трябва да се дефинират и анализират чрез изрични процеси. Системата трябва да бъде защитена от нежелано поведение на нейните системни елементи чрез изричното дефиниране на тяхното желано поведение като договор между заинтересованата система и всеки неин системен елемент. Договорът трябва да бъде ясен и машинно изпълним (цифров договор) с истински процеси и правила. Договорите, от своя страна, също трябва да бъдат защитени. Постоянният мониторинг на всички елементи на системата е задължителен.

– Анализът с цел предсказване на всички елементи на системата по отношение на сигурността е силно желателен. Този анализ включва: Предположения за риска (напр. предположения относно заплахите, уязвимостите, последствията/въздействието и вероятността от възникване, които влияят върху начина, по който рискът се оценява, реагира и наблюдава във времето);

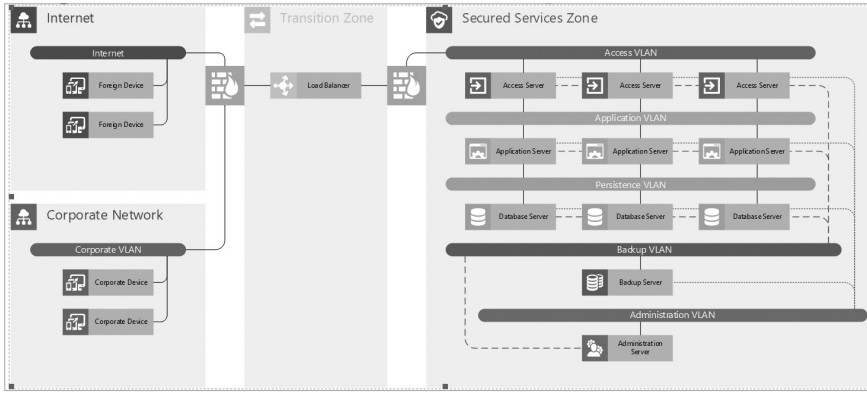
– Ограничения на риска (напр. ограничения върху разглежданите алтернативи за оценка на риска, реакция и мониторинг);

– Толерантност към риска (напр. нива на риск, видове риск и степен на несигурност на риска, които са приемливи); и

– Приоритети и компромиси (напр. относителната важност на мисиите/бизнес функциите, компромиси между различните видове риск, пред които са изправени организациите, времеви рамки, в които организациите трябва да се справят с риска, и всички фактори на несигурност, които организациите вземат предвид при реагирането на риска) (Petrova, 2021).

Едно от възможните решения е минимизиране на надеждните мрежови зони (фиг. 10) чрез „вътрешни“ облаци (всяко приложение може да бъде в такъв „вътрешен“ облак); всички участници

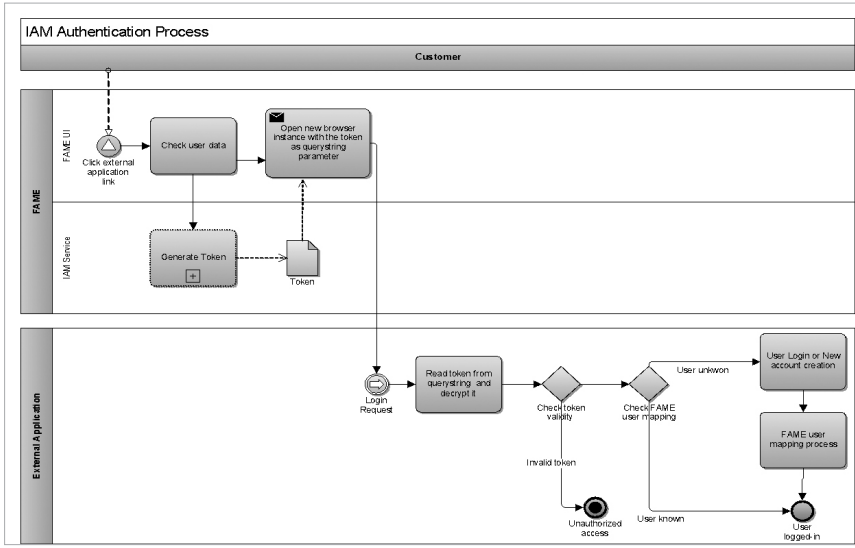
в мрежата се свързват от интернет (т.е. крайните точки се третират като интернет възли). Една от причините за това е, че заплахите от вътрешни и външни лица са еднакво опасни (http://www.visualcapitalist.com/cybersecurity-threat-insajders-outsiders/).



Фигура 10. Пример за разделени интернет зони

Референтната информация между всички системни елементи и свързани системи трябва да се синхронизира. Блокчейн технологиите могат да бъдат разгледани като потенциално решение на този проблем за случаите, когато съхранението и достоверността на информацията е от критично значение.

Трябва да бъдат проектирани услуги за всички актьори в системата (хора, организация, групи, инструменти). За всеки от актьорите трябва да бъдат определени: записване (и информацията, която се съхранява за съответния актьор), удостоверяване (потвърждаване на искове), идентификация (не е задължителна) и всички бизнес процеси от техния жизнен цикъл.

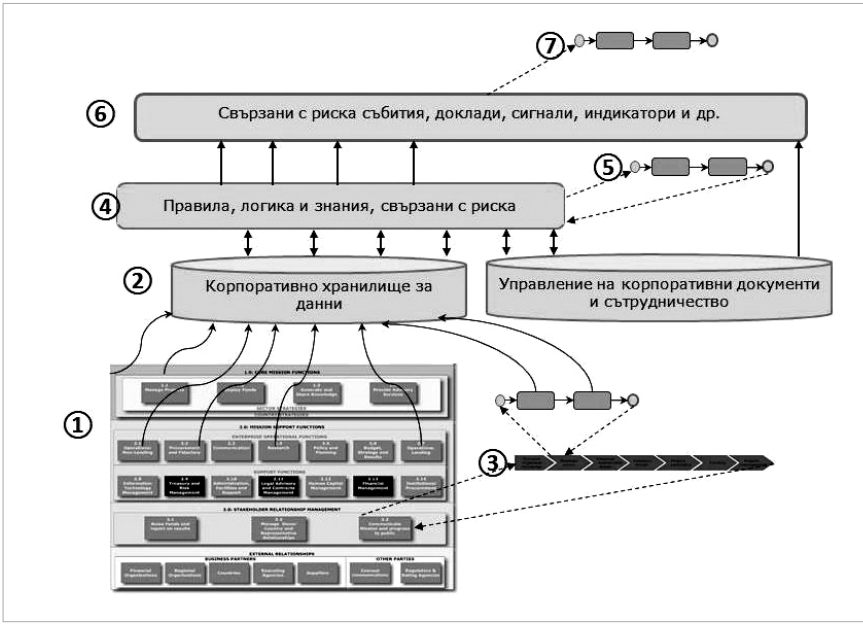


Фигура 11. Модел на процес за удостоверяване

Услугите за управление на достъпа включват: дефиниране на ролите, определяне на членовете на всяка група, дефиниране на операциите (функциите) на всяка от групите, определяне на бизнес обектите. Ролята се дефинира като набор от отговорности, получени като права (правомощия за изпълнение) или задължения (необходими за извършване като част от работата).

Отговорността се състои от два взаимосвързани елемента: разрешение за изпълнение на определена операция върху определен обект с определени параметри и забрана за изпълнение на същата операция върху същия обект с определени параметри. Въпросите, на които трябва да бъдат намерени отговори са: КОЙ (роли) може / не може да прави КАКВО (операции) с КАКВО (обекти) КОГА (време) и КЪДЕ (местоположение). Примери за различни роли, които може да бъдат дефинирани са: организационна роля (мениджър), организационна група (всеки от отдел), йерархична група (всички мениджъри), функционални групи, роли на процеса (собственици на процеси, инициатори на процеси), роли на дейности (изпълнители на дейност, супервайзори на дейност), експертни групи, роли в проекта, група за сигурност и др.

Използването на бизнес процеси активира сигурността и контрола на риска.



Фигура 12. Интегриране на управлението на риска с бизнес процесите

Рискът трябва да бъде внимателно наблюдаван, оценяван и спрямо темпото на бизнес процесите да се предприемат необходимите действия. На фигура 12 е представена връзката между бизнес процесите и управлението на риска. На фигурата са означени следните елементи:

1. Бизнес функциите на предприятието трябва да бъдат обогатени, за да генерират данни, свързани с риска.

2. Тези данни, свързани с риска, трябва да се събират в хранилището на корпоративни данни заедно с други бизнес данни.

3. Някои бизнес процеси трябва да бъдат актуализирани, за да вградят дейности, свързани с риска.

4. Набор от свързани с риска правила, логика и знания, свързани с риска, трябва да могат да използват свързаните с ри-

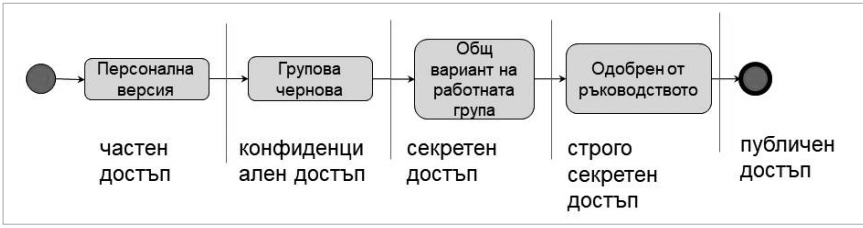
ска и други бизнес данни за откриване на приемливи граници на риска, както и взаимозависимости и корелации между различни рискове.

5. Някои бизнес процеси за намаляване на риска може да се активират автоматично.

6. Много показатели, свързани с риска, сигнали трябва да са налични под формата на табла за управление и отчети, достъпни за различни членове на персонала.

7. Членовете на персонала трябва да могат да инициират бизнес процеси въз основа на наблюдаваната информация, свързана с риска.

Правата за достъп трябва да са съобразени с динамиката на работата, която трябва да се свърши. Например, да се съгласува нивото на сигурността на бизнес обект (напр. организационен документ) с напредъка на работата (изготвяне на този документ). Статично разделяне на задълженията (едно и също лице не трябва да изпълнява „DO“ и „VALIDATE“).



Фигура 13. Динамична промяна на достъпа до даден обект в зависимост от фазата на процеса

2. Сигурност по дизайн при разработка на уеб приложения

OWASP (Open Web Application Security Project) е онлайн общност, която разработва безплатни инструменти, документация, статии и технологии, за да помогне на специалистите при защитата на уеб сайтове, уеб приложения и мрежови ресурси. Тя е основана от Марк Кърфи, опитен специалист по информационна сигурност, през 2001 г. Техният основен фокус е върху уеб сигурността, сигурността на приложенията и оценката на уязвимостите.

OWASP са разработили и собствени принципи за проектиране на сигурността, за да помогнат на разработчиците да изградят

силно защитени уеб приложения (https://patchstack.com/articles/security-design-principles-owasp/).

Принципите на проектиране на сигурността на OWASP са следните:

1. Идентифициране на активите.

Преди да се разработи стратегии за сигурност, от съществено значение е да идентифицират и класифицирате данните, които приложението ще обработва. OWASP предлага на програмистите да създадат контроли за сигурност, които са подходящи за стойността на управляваните данни. Например, приложение, обработващо финансова информация, трябва да има много по-строги ограничения от блог или уеб форум.

2. Разбиране на нападателите.

Програмистите трябва да проектират контроли, които предотвратяват злоупотреба с приложението от различни видове злонамерени страни, включително (от най-много до най-малко опасни). Могат да бъдат обособени няколко основни групи нападатели:

– Недоволни служители и програмисти;

– Случайни Drive-by атаки, които пускат вируси или троянски атаки върху системата;

– Мотивирани киберпрестъпници;

– Престъпни организации със злонамерени цели;

– Учещи се, които искат да изпробват възможностите.

Най-опасният тип атаки, от които разработчиците трябва да се предпазят, са от недоволни членове на персонала и програмисти. Това е така, защото те обикновено имат високо ниво на достъп до чувствителни системи. Програмистите могат да използват техники на принципите на OWASP, за да се предпазят от тези видове атаки.

Особено внимание към основни стълбове на информационната сигурност.

OWASP препоръчва всички контроли за сигурност да бъдат проектирани с оглед на основните стълбове на информационната сигурност:

– Поверителност – разрешаване на достъп само до данни, за които потребителят има разрешение;

– Цялост – гарантиране, че данните не са подправени или променени от неоторизирани потребители

– Наличност – гарантиране, че системите и данните са достъпни за оторизирани потребители, когато имат нужда от тях.

2.1. Архитектура на сигурността

OWASP препоръчва всяко приложение да има мерки за сигурност на приложението, предназначени да покриват всички видове рискове, вариращи от типични рискове при използване (случайно изтриване на данни) до екстремни атаки (атаки с груба сила, атаки чрез инжектиране и т.н.). Те препоръчват разработчиците да обмислят всяка функция на приложението, което проектират, и да зададат следните въпроси:

– Процесът около тази функция достатъчно ли е безопасен? С други думи, това правилен процес ли е?

– Ако бях недоброжелател, как бих могъл да злоупотребя с тази функция?

– Изисква ли се функцията да е включена по подразбиране? Ако е така, има ли ограничения или опции, които биха могли да помогнат за намаляване на риска от тази функция?

Чрез „мислене за вреда“ разработчиците могат да идентифицират начините, по които киберпрестъпниците и злонамерените лица могат да се опитат да атакуват уеб приложение.

OWASP предлага разработчиците също да следват техниката за моделиране на риска от заплахи STRIDE / DREAD, използвана от много корпорации. STRIDE помага на програмистите да идентифицират заплахите, а DREAD позволява на програмистите да оценяват заплахите.

3. Принципи на сигурността

Тези принципи се съдържат в Ръководството за разработка на OWASP и са в съответствие с принципите за сигурност, изложени в книгата на Майкъл Хауърд и Дейвид ЛеБланк „Писане на защитен код“.

Те включват:

1. Минимизиране на повърхността на атаката

Всеки път, когато програмист добави функция към своето приложение, той увеличава риска от уязвимост на сигурността. Принципът за минимизиране на площта на атаката ограничава функциите, до които потребителите имат достъп, за да се намалят потенциалните уязвимости. Например, ако трябва да се кодира функция за търсене в приложение, тази функция за търсене е потенциално уязвима за атаки за включване на файлове и атаки за инжектиране на SQL. Разработчикът може да ограничи достъпа до функцията за търсене, така че само регистрирани потребители да могат да я използват, намалявайки повърхността за атака и риска от успешна атака.

2. Установяване на сигурни настройки по подразбиране

Този принцип гласи, че приложението трябва да е защитено по подразбиране. Това означава, че нов потребител трябва да предприеме стъпки за получаване на по-високи привилегии и премахване на допълнителни мерки за сигурност (ако е позволено). Установяването на безопасни настройки по подразбиране означава, че трябва да има строги правила за сигурност за това как се обработват регистрациите на потребители, колко често трябва да се актуализират паролите, колко сложни трябва да бъдат паролите и т.н. Потребителите на приложението може да могат да изключат някои от тези функции, но те трябва да бъдат настроени на високо ниво на защита по подразбиране.

3. Принципът на най-малката привилегия

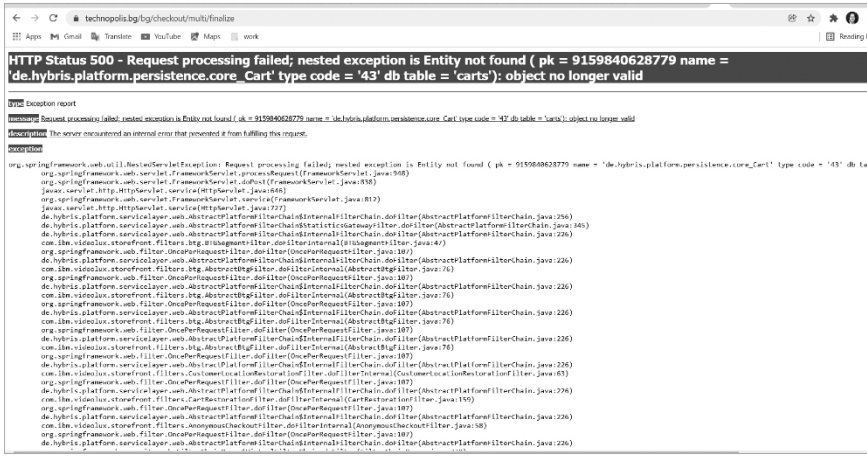
Принципът на най-ниската привилегия (POLP Principle of Least Privilege) гласи, че потребителят трябва да има минималния набор от привилегии, необходими за изпълнение на конкретна задача. POLP може да се приложи към всички аспекти на уеб приложение, включително потребителски права и достъп до ресурси. Например, потребител, който е регистриран в блог приложение като „автор“, не трябва да има административни привилегии, които му позволяват да добавя или премахва потребители. Те трябва да имат право само да публикуват статии в приложението.

4. Принципът на защитата в дълбочина

Принципът на защита в дълбочина гласи, че множеството контроли за сигурност, които подхождат към рисковете по различни начини, са най-добрият вариант за защита на приложение. Така че, вместо един контрол за сигурност за потребителски достъп, по-добре да имаме множество слоеве за валидиране, допълнителни инструменти за проверка на сигурността и инструменти за регистриране. Например, вместо да се позволи на потребител да влезе само с потребителско име и парола, може да се използвали проверка на IP, система Captcha, регистриране на броя опити за влизане, откриване чрез груба сила и т.н.

5. Принципът за сигурни провали

Има много причини, поради които уеб приложение не може да обработи транзакция. Може би връзката с база данни е неуспешна или данните, въведени от потребител, са неправилни. Този принцип гласи, че приложенията трябва да се провалят по сигурен начин. Неуспехът не трябва да дава на потребителя допълнителни привилегии и не трябва да показва на потребителя чувствителна информация като заявки към база данни или регистрационни файлове. Това означава, че всички възможни провали трябва да бъдат предвидени и при поява грешките обработени. На следващата фигура 14 е показан пример за нарушаване на това правило.



Фигура 14. Пример за „изтичане“ на информация в резултат на необработване на грешка

6. Принцип за недоверие по отношение на услугите

Много уеб приложения използват услуги на трети страни за достъп до допълнителна функционалност или получаване на допълнителни данни. Този принцип гласи, че никога не трябва да се доверявате на тези услуги от гледна точка на сигурността. Приложението трябва винаги да проверява валидността на данните, които услугите на трети страни изпращат, и да не дава на тези услуги разрешения на високо ниво в приложението.

7. Принцип за разделяне на задълженията

Разделянето на задълженията може да се използва за предотвратяване на измамни действия на лица. Например, потребител на уебсайт за електронна търговия не трябва да бъде повишен в администратор, тъй като той ще може да променя поръчките и да си дава продукти. Обратното също е вярно – администраторът не трябва да има възможност да прави неща, които правят клиентите, като например да поръчва артикули от потребителската страна на уебсайта.

8. Принцип за избягване на сигурността чрез неизвестност

Този принцип на OWASP гласи, че никога не трябва да се разчита на сигурност чрез неизвестност. Ако приложение изисква административният URL адрес да бъде скрит, за да може да остане защитено, тогава то изобщо не е защитено. Трябва да има достатъчно средства за контрол на сигурността, за да се запази приложението безопасно, без да се крие основната функционалност или изходния код.

9. Принцип за поддържане на простота в сигурността

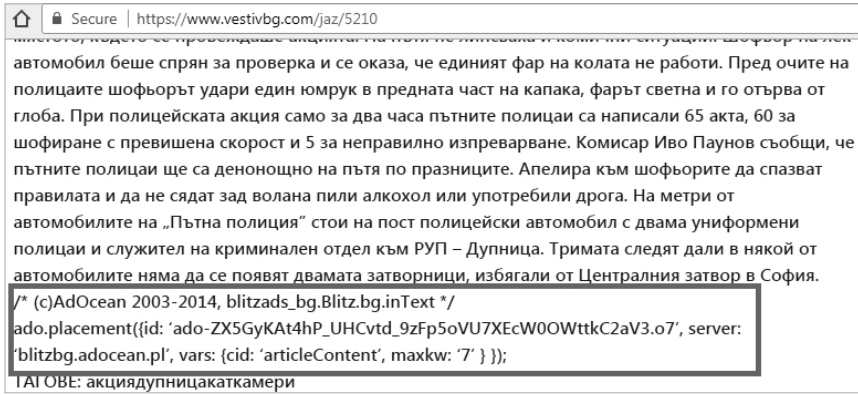
Разработчиците трябва да избягват използването на много сложна архитектура, когато разработват контроли за сигурност за своите приложения. Наличието на много сложни механизми може да увеличи риска от грешки.

10. Принцип за правилно коригиране на проблемите със сигурността

Ако в приложение е идентифициран проблем със сигурността, разработчиците трябва да определят основната причина за проблема, т.е. така наречената „коренова причина“ и де се опитат

да отстранят нея а не последствията от нарушената сигурност. Например, при изтичане на паролите на потребителите очевидната първа стъпка е смяна на паролите, но това не премахва основната причина и възможността от ново изтичане. За да се реши проблема трябва да бъде намерено слабото място в системата, довело до изтичането и то да бъде премахнато. След поправката програмистите трябва да тестват ремонта щателно. Ако приложението използва шаблони за проектиране, е вероятно грешката да присъства в множество системи. Програмистите трябва да внимават и да идентифицират всички засегнати системи.

Един от основните проблеми със сигурността при разработката на приложения, според OWASP е лошото кодиране (фиг. 15).



Фигура 15. Пример за „изтичане" на информация в резултат на грешка при програмирането

Повечето програмисти се придържат към определени стандарти, когато кодират софтуерни приложения. Тези стандарти са предназначени да гарантират, че приложението е лесно за поддръжка, защитено и работи добре. За съжаление, някои програмисти са по-малко квалифицирани, мързеливи, невнимателни или работят в кратък срок, което води до грешки в техния код. Някои от тези грешки увеличават риска от уязвимост, която може да бъде използвана от хакери (Petrova, 2022). Най-често срещаните лоши практики за кодиране включват:

1. Не добро форматиране на кода

Добрите програмисти форматират своя код по начин, който е лесен за четене. Те поддържат широко използван стандарт за форматиране в цялата си работа, така че другите програмисти да могат да го разберат лесно и да пишат код в същия стил. Лошото използване на форматиране се счита за лошо кодиране, защото прави кода по-труден за поддръжка.

2. Печатни грешки в кода

Програмистите понякога грешно изписват имена на променливи или функции. Тази проста грешка може напълно да спре приложението да работи или да създаде уязвимост, която може да бъде използвана. За съжаление, тази грешка не винаги може да бъде открита и, както например в следващата фигура да изведе на потребителя нежелана информация за кода (фиг. 16).

Фигура 16. Информация за базата данни в резултат на програмистка грешка

3. Неизползване на модулен код

Модулният код разделя логиката на програмата на класове и функции. Улеснява разбирането на логиката на приложението и съкращава отделните файлове. Лошото кодиране би било големи участъци от код, съдържащи цикли, многобройни оператори „if“ и обърквaщи участъци от кода. Немодулният код става по-труден за поддръжка, което води до повече уязвимости.

4. Лошо обработване на въвежданите от потребителя данни

Обработването на входа и съхраняването на данните, предоставени на приложението от потребителите са една от основните задачи при кодиране на приложението. Това е така, защото потребителските данни често са източник на кибератаки, включително SQL инжекции и Cross-site Scripting (XSS). Лошите практики за кодиране няма да обработят потребителските данни по подходящ начин, добавяйки уязвимости към приложението.

5. Недостатъчна защита на данните

Лошото кодиране включва практики като твърдо кодиране на пароли във файлове и неправилно използване на криптиране. Тези практики правят данните на приложението по-уязвими за хакване. В допълнение към това правило трябва да кажем, че процесът по кодиране трябва да се разглежда и във времето, като постоянно се следи кои са актуалните методи за кодиране и кои кодове са вече разбити и не трябва да бъдат използвани, т.е. освен че принципно трябва да бъде използвано кодиране на тези данни, то е необходимо то да е актуално към съответния момент и при смяна на ситуацията (разбиване на кода) то да бъде заменено.

6. Недостатъчно регистриране

В идеалния случай приложенията ще регистрират значими събития, които се случват, включително влизане на потребители и транзакции в базата данни. Лошите практики за кодиране не включват регистриране, което прави много по-трудно проследяването на опити за хакване или уязвимости.

7. Лоша обработка на грешки

Лошите практики за кодиране могат да разкрият информация за грешки, която е полезна за хакерите, както у показано на фигури 14, 15 и 16. Грешките трябва да се обработват елегантно и по начин, който минимизира излагането на информация за приложението пред потребителя.

Според Института за софтуерно инженерство (Ettinger, 2019) 90 процента от всички докладвани инциденти със сигурността възникват в резултат на експлойти и дефекти в дизайна или кода на софтуера. Много от тези дефекти са възникнали в резултат на лоши

практики за кодиране. Според доклада Cyber Risk на HP Security Research за 2015 г. (HP, 2015): „Основните причини за често експлоатирани софтуерни уязвимости са постоянно дефекти, грешки и логически грешки. Специалистите по сигурността са открили, че повечето уязвимости произтичат от сравнително малък брой често срещани грешки в софтуерното програмиране. Ясно е, че лошите практики за кодиране могат да създадат уязвимости, които могат да бъдат използвани от киберпрестъпниците.

Като пример можем да бъде разгледана практиката при програмиране на WordPress, като една от най-често използваните към момента среди за масово създаване на сайтове и несложни уеб приложения, включително и онлайн магазини.

1. Конвенциите за именуване на WordPress са странни.

Една от най-честите критики към WordPress е неговите необичайни и неправилни имена на променливи и функции. Например има функция get_the_content(), която връща данни за публикации в блогове, и функция the_content(), която отпечатва данни за публикации в блогове. Но странното е, че get_permalink() връща данни – когато бихте очаквали тази функция да се извика get_the_permalink(), ако наименуването е последователно. Това демонстрира липсата на внимание и наследените грешки, присъстващи в ядрото на WordPress.

2. В WordPress липсват някои много основни функции

WordPress е много гъвкава система за управление на съдържанието, която може да се използва за всичко – от мощни магазини за електронна търговия до прости блогове, въпреки това най-полезните функции в приложенията на WordPress обикновено се постигат чрез използването на плъгини и персонализирани теми. В основното приложение също липсват някои основни функции в сравнение с други приложения. Една част от функционалността, която липсва, е разширеното кеширане. Необходимо е да бъдат инсталират добавки на трети страни, за да е възможно кеширане на паметта, кеширане на диска и минимизиране на файлове. Изискването да се инсталира софтуер на трети страни за това, което е основна функция в други системи за управление на съдържанието,

е необичайно. Това, също така, увеличава риска от уязвимости, тъй като повечето плъгини са кодирани от хора, които не са наети от WordPress и са под по-слаб надзор, в сравнение с разработчиците на ядрото на WordPress.

3. WordPress използва някои стари техники за кодиране

WordPress е една от малкото съвременни системи за управление на съдържанието, които разчитат на функции, ограничен брой класове и много кодови цикли. Приложението също е смесица от HTML, Javascript и CSS. Този подход е донякъде остарял, като повечето модерни CMS приложения използват дизайн на приложения от вида модел-изглед контролер (MVC) и стриктно прилагане на обектно-ориентирано програмиране (OOP). MVC е много по-чист и лесен за поддръжка, което го прави по-безопасен за използване.

4. Приставките и темите на трети страни са източник на безпокойство

Тъй като потребителите на WordPress често са принудени да инсталират плъгини на трети страни, за да получат определена функционалност, е по-вероятно те да бъдат изложени на лошо кодиране. Ниските стандарти за кодиране в някои от тях, наблюдавани от специалисти, ясно са демонстрирани от големия брой уязвимости, открити в добавките. Според популярните бази данни за докладване не уязвимости от 3972 известни уязвимости на WordPress, 52% са от плъгини на WordPress, 37% от ядрото на WordPress и 11% от теми на WordPress. Много плъгини са уязвими към експлойти за включване на файлове и SQL инжекции. За съжаление, някои добавки остават в несигурно състояние месеци или години, преди да бъдат коригирани.

Въпреки проблемите с ядрото на WordPress, плъгините на трети страни и темите на трети страни, WordPress остава една от най-безопасните системи за управление на съдържанието. Тъй като това е проект с отворен код, има хиляди разработчици и сътрудници по целия свят, които търсят грешки и уязвимости в кодовата база. Екипът за разработка на WordPress е много проактивен, когато става въпрос за идентифициране и отстраняване на грешки и уязвимости в ядрото на WordPress. Те са описали подробно техния

подход към защитата на WordPress и са много бързи, когато става въпрос за коригиране на проблеми със сигурността.

Екипът за разработка непрекъснато укрепва приложението, за да се справи с често срещаните проблеми със сигурността, включително тези, изброени от Open Web Application Security Project (OWASP). Тези проблеми включват атаки с груба сила, инжектиране на файлове и SQL инжектиране.

Атаката с груба сила brute-force attack е опит за откриване на парола за валиден потребителски акаунт чрез използване на предварително зададени стойности. Най-често срещаният пример е речниковата атака. Атаките с речник често са успешни, защото много хора са склонни да използват кратки пароли. Други форми на атака с груба сила може да опитат комбинации от букви и цифри. Автоматизираният софтуер често се използва за отгатване на хиляди комбинации от пароли.

Хакерите използват атаки с груба сила, за да проникнат в сайтове, след което да ги компрометират, като качват зловреден софтуер чрез редактора на теми или плъгини.

Повече информация и подробности за механизма на атаките по отношение на Word Press сайтове с кодове може да бъде открита на адрес https://patchstack.com/articles/brute-force-attack/ а относно инфилнриране на червей с цел блудфорс атака в Word Press на адрес https://themeisle.com/blog/malware-in-wordpress/.

Има няколко причини хакерите да се насочат към атака на уеб сайт:

– искат да получат достъп до данните на сайта (евентуално да ги изтеглят и използват за други цели, или да ги манипулират);

– искат да включат реклами, изскачащи прозорци, пренасочвания и друг зловреден софтуер;

– искат да го използват за изпращане на нежелана поща до потребители, които имат доверие на сайта.

Съществуват различни начини да бъдат предотвратени такъв вид атаки и те са добре известни:

1. Ограничаване на опитите за влизане – може да защити сайта срещу груби атаки, като се ограничат неуспешните опити за влизане до определен брой в рамките на фиксирано време. Това ще

забави много във времето и като резултат ще обезмисли атаката, защото тя разчита на налучкване чрез многобройни опити.

2. Добавяне на Captcha – по принцип ботовете не са в състояние да разгадаят captcha и това помага да се забавят опитите за груба сила. Може да се използва Patchstack, за добавяне на невидим captcha към сайта.

3. Ограничаване на достъпа до страницата за вход в WordPress – друг начин за защита на WordPress сайт срещу атаки с груба сила е чрез ограничаване на достъпа до страницата за администриране. Например, когато се използва CAPTCHA защита, ресурсите на сървъра все още се използват за извличане и показване на страницата за вход в WordPress. Можете да се ограничи достъпа до /wp-login.php само до конкретно IP чрез .htaccess файла.

4. Използване на силни пароли – сигурността на паролата често се пренебрегва. Процентът на успеваемост при груба атака зависи от дължината и сложността на паролата.

5. Деактивиране на файловия редактор – добра практика е да се деактивира файловия редактор в WordPress, което би попречила на атака чриз инфилтриране на код във файла 404.php. За да се деактивира редактора, трябва да се добави define('DISALLOW_FILE_EDIT', true); към файла wp-config.php.

6. Принципът на най-малко привилегии (POLP) и различни потребителски роли в WordPress – поддържането на високо ниво на сигурност е съществена част от управлението на успешен уе бсайт на WordPress. Неспазването на това може да доведе до хакване на уеб сайта, причинявайки огромни щети на репутацията на бизнеса и потенциално да доведе до загуба на приходи. Ключов компонент на сигурността на WordPress, който често се пренебрегва, са потребителските привилегии. Привилегиите са функциите, до които всеки потребител на WordPress има достъп. Случайното предоставяне на достъп на потребители на ниско ниво до грешни права може да доведе до нарушаване на сигурността или загуба на данни. Най-добрият начин за правилно конфигуриране на привилегиите в WordPress е да се приложи принципа на най-малката привилегия (POLP). Това е принцип на сигурност, който се използ-

ва, за да се гарантира, че потребителите няма да получат достъп до функционалност, която не им е абсолютно необходима.

Принципът на най-малката привилегия гласи, че потребителят трябва да има достъп само до информацията и ресурсите, които са абсолютно необходими за дефинирана цел. Когато този принцип се прилага към типове потребителски акаунти (администратор, абонат, автор), това означава, че на всеки вид потребителски акаунт трябва да се дават само привилегиите, които са от съществено значение за изпълнение на предназначената му функция, т.е. необходимите за изпълнение на задачите в рамките на бизнес процесите, в които участва, или е отговорен. Така че, ако съществува роля на автор за хора, които пишат публикации в блогове, тази роля трябва да позволява на потребителя само да получи привилегиите, необходими за да бъде автор – включително преглеждане, създаване, модифициране и изтриване на собствени публикации. Дефинираните права не трябва да позволяват достъп до други привилегии на високо ниво, като промяна на паролата на администратора, например. Когато принципът на най-малките привилегии се прилага към потребителите, това означава, че на всеки потребител трябва да бъде възложен само възможно най-малкият достъп за действието, което трябва да извърши. Например, на служител, който ще пише публикации в блогове, трябва да бъде присвоена само ролята на автор, а не роля на администратор. Същият принцип се прилага за технически въпроси, свързани с приложение, като разрешенията на потребител на база данни и разрешенията за файлове на приложението. Потребителите трябва да имат достъп само до привилегиите, които са им абсолютно необходими във връзка с осъществяване на задълженията.

POLP също така заявява, че привилегиите трябва да се предоставят само за времето, когато действието е необходимо. Така че, ако се предвижда възможността за гост-автори, които ще напишат една публикация в блога, трябва да се предвиди процедура за премахване на техните привилегии за автор, след като завършат статията. Няма причина те да запазят тази роля, след като действието приключи. Много е важно при такава ситуация и да се разпределят правилно ролите и задълженията в рамките на предвидената процедура за това и тя ясно да дава отговор на въпросите от вида:

– Кой и при какви обстоятелства заявява даването на съответните права на нов потребител;

– Кой реално извършва операцията по присвояване на правата;

– Кой и при какви обстоятелства заявява премахването на права;

– Кой и в какви срокове премахва или променя дадените права.

Най-добрият вариант в случая е да съществуват предварително дефинирани роли, които собствениците на съответния бизнес процес имат право да назначават на съответните изпълнители. В противен случай, при необходимост от включването на администратор, процесът може да стане ненужно бавен а участието на допълнителен субективен фактор го прави и по-несигурен. При недобре дефиниран и документиран процес може да възникнат противоречия между замесените страни, както и проблеми при одит.

Ползите от прилагането на POLP към приложението включват:

1. По-добра сигурност на системата

Гарантирането, че потребителите нямат прекомерни привилегии, може да помогне за намаляване на риска от вътрешни заплахи (недоволни или злонамерени служители). Те няма да имат високо ниво на достъп, така че не могат лесно да изтрият съдържание, или да откраднат информация. От друга страна, прилагането на POLP означава, че хакерите няма да имат незабавен достъп до привилегии на високо ниво, ако успеят да хакнат акаунта на потребител. Освен това има по-малък риск потребителите случайно да повредят уеб приложението, когато щракнат върху грешен бутон.

2. По-лесна потребителска поддръжка

По-лесно е да се управляват потребители когато има по-малко потребители с привилегии на високо ниво. Извършването на операции, изискващи висок приоритет ще е по-лесно и уведомяването на засегнатите потребители за промените ще бъде по-лесно.

3. Използване на роли за прилагане на POLP

Ролите са удобен начин за присвояване на привилегии на различни потребители. Те ускоряват процеса и не позволяват на

потребителите да извършват действия, до които не трябва да имат достъп. WordPress, например, разполага с шест предварително дефинирани роли, с предварително зададен набор от права (привилегии). Те включват:

– Роля на администратор – администраторът е ролята, възложена на лицето, което инсталира WordPress. Той получава пълни права и може да извършва всякакви действия.Администраторите могат да добавят или премахват потребители, да добавят или премахват съдържание, да променят теми, да инсталират или премахват добавки, да модерират коментари и много повече. И дори и тези права трябва добре да се обмислят, защото при поддъжането на блог, например е резонно да се запитаме, трябва ли администраторът, който е чисто техническо лице да има права да управлява мненията на потребителите на блога, или това трябва да е приоритет само за модераторите.

– Роля на супер администратор – супер администраторите съществуват само в многосайтови инсталации на WordPress, където една инсталация се използва за стартиране на много отделни уебсайтове. Те имат права да добавят или премахват администратори, да добавят или премахват блогове и страници, да преименуват мрежата от сайтове и да променят темите или добавките, които администраторите могат да използват.

– Роля на редактор – редакторите има роля на мениджъри на съдържанието в уеб сайтовете. Те имат права да пишат, редактират и изтриват публикации, написани от други потребители. Те могат също да пишат, редактират и изтриват коментари, написани от други хора, да променят категории, да четат частни публикации и съобщения, да управляват тагове и да създават персонализирани таксономии. Основното ограничение на ролята на редактор е, че те не могат да променят настройките на сайта, потребителските роли, темите и добавките. Тъй като редакторите могат да изтрият всяка публикация на уебсайта, тази роля трябва да се дава само на ограничен брой, отговорни за тази дейност лица.

– Авторска роля – авторите имат роля на създател на съдържание в уебсайтовете на WordPress. Те могат да качват файлове, да пишат, редактират, публикуват и изтриват свои собствени статии.

Могат да променят данните в своя потребителски акаунт, включително име, аватар, биография и парола. Не могат да редактират публикации на други потребители и нямат достъп до функционалност за администриране на по-високо ниво.

– Роля на сътрудник – сътрудникът е подобен на автор. Съществената разлика е, че те не могат да изтрият собствените си публикации, след като са били публикувани. Това е полезна роля, защото не позволява на недоволните служители да изтрият работата си, ако бъдат уволнени.

– Роля на абонат – ролята на абонат има много ограничени възможности. Абонатите имат право само да създават и променят своя личен профил и да оставят коментари. Основната цел на тази роля е да улесни потребителите да оставят коментари, без да е необходимо да са влезли в профила си.

Често срещана грешка, допускана от собствениците на уеб сайтове, е да дадат на всички представители на организацията ролята на администратор. Това може да изглежда най-лесния начин да позволят на всеки да си свърши работата, но за съжаление, това дава възможност на недоволен служител да повреди сайта и прави хакването на потребителски акаунти много по-опасно. В този случай потребителите, макар и вътрешни за организацията, притежател на сайта, могат да причинят много повече щети ако направят грешка в секцията за администриране.

4. Потребителски привилегии по отношение на базата данни (по примера на WordPress)

Принципите на POLP трябва да се прилагат и към базата данни, част от самото приложение WordPress (Agboola, 2022). Веднъж инсталиран, WordPress трябва да се нуждае само от възможността да чете, пише, актуализира и изтрива данни от базата данни. Потребителят на база данни на WordPress не се нуждае от разрешение за добавяне на потребители на база данни, премахване на бази данни, промяна на структурата на базата данни и т.н.

Разрешенията на сървъра, свързани с WordPress файлове и директории, също трябва да бъдат ограничени. Това се постига на

ниво сървър. Предотвратява определени типове злонамерени атаки, включително включвания на злонамерени файлове.

Специализирана база данни с уязвимости на WordPress е налична на адрес https://patchstack.com/database/

WordPress поддържа ръководство за подобряване на сигурността на WordPress, което включва информация за правилните разрешения за файлове – https://wordpress.org/support/article/hardening-wordpress/

Възможно е да има ситуации, в които са необходими множество администратори с високо ниво на достъп. Въпреки това, човекът, който е основал сайта, все пак може да се наложи да ограничи достъпа до чувствителните данни и функционалността на добавките. Плъгините, които използват чувствителни данни, обикновено могат да бъдат конфигурирани да изключват конкретни потребители, включително администратори. Повечето уебсайтове на WordPress са настроени с достъп до протокол за прехвърляне на файлове (FTP). Това позволява на потребителите да качват файлове директно на сървъра. FTP поддържа потребителски роли, така че в повечето случаи е необходимо да се изключи правото на потребителите да качват файлове в конкретни директории в съответствие с POLP.

Вероятно не е изненада, че хората използват лоши пароли. Скорошно проучване на публично достъпни „хакнати“ акаунти разкрива, че „123456“ е най-използваната парола, следвана от „много по-сигурната“ „123456789“ и „трудната за отгатване“ „qwerty“. Проучване показва, че има повече от половин милион случая, когато футболни (или футболни) фенове използват имената на клубовете „Ливърпул“ или „Челси“ като свои пароли.

5. Софтуер за управление на паролите

Потребителите на софтуер в общия случай не харесват паролите и не обичат да генерират нови пароли. Това е причината, поради която трябва да се използват инструменти за управление на пароли. Добра практика би била приложението, освен да изисква от потребителите генериране на парола с достатъчно ниво на сложност, което да проверява при регистрация да препоръчва и използването на софтуер за управление на паролите.

Няколко са основните причини за използване на софтуер за управление на паролите:

● Не е възможно да бъдат запомнени всички пароли за множеството използване всеки ден приложения. Лоша, макар и често използвана практика е използването на една и съща парола за повече от един акаунт. С използване на инструментите за управление на пароли е възможно получаването на достъп до всички пароли от едно място с един главен ключ.

● Използването на инструмент за управление на паролите позволява генерирането на произволен ключ, който да отговаря на изискванията на приложението и неговото съхранение.

● От съществено значение е главният ключ, тъй като той отговаря за съхранението на всички пароли. Би трябвало да се използва парола, която е много по-дълга, като се използват цифри, главни и малки букви. Един от добрите варианти е генерирането на кратко изречение, но трябва да е нещо важно, за за бъде запомнено без проблеми. От друга страна не трябва да е свързано с публично достъпна информация, която може лесно да бъде открита.

● Примери за приложение за управление на пароли са LastPass и KeePass, като LastPass се използва широко, има добър потребителски интерфейс и функционалност за управление на пароли за много потребители. Други подобни приложения са Dashlane и 1Password. От своя страна Chrome предлага тази функционалност по отношение на паролите за уеб приложения, достъпвани чрез него. Недостатък е, че потребителите рядко използват достатъчно силни пароли за google акаунта си, който на практика е ключ към всички останали пароли. Затова при работа с приложения от висока важност и чувствителна информация е по-добре да се използва специализирано приложение.

6. Сигурност чрез неизвестност/секретност

Известен като STO (Security Through Obscurity), „сигурност чрез неизвестност“ или „секретност“, това е един от начините, по които системните инженери и разработчиците на софтуер осигуря-

ват защита на система или приложение. STO се основава предимно на скриване на важна информация и налагане на секретност като основна техника за сигурност. Използването на защита чрез секретност се разглежда като средство за свеждане до минимум на риска от атака. Най-лесно можем да обясним принципа на работа на сигурността чрез неизвестност с пример от реалния живот. И първият пример, който идва на ум е този: скриване на ключа от входната ви врата под близката саксия или постелката за пред вратата. Принципът е прост: къщата ще бъде „сигурна“, докато крадец не открие ключа в скривалището му, след което къщата става уязвима.

В света на киберсигурността има други сценарии, които демонстрират практики за сигурност чрез неизвестност: скриване на потребителски пароли в двоичен код или смесени със скрипт код или коментари. Това е много популярна техника, която предполага, че нападателят няма да прочете кода и следователно осигурява защита от всяко проникване.

Промяна на името на важни папки на приложението, например от „admin“ на „_admin.“ Това може да забави атакуващия, но ако той открие смененото на „_admin“ име на папката и няма допълнително удостоверяване или базиран на IP бял списък, ще може да получи достъп направо до административната част на приложението. Същото се отнася и за системните таблици в базите данни. Независимо как е кръстена таблицата, съхраняваща акаунти и пароли на потребителите ако съхраняваме паролите в некриптиран вид, или криптирани с компрометиран код, това няма да спре атакуващата страна.

Използването на различен порт на демон е много популярна техника за намаляване на количеството груби атаки срещу определени портове, като порт 22 на SSH. Тази техника работи; обаче, след като нападателят открие, че SSH работи на порт, различен от 22, той така или иначе ще започне да се насочва към новия порт. Подходящо решение би било да се деактивира удостоверяването с парола и да се ограничат влизанията по IP с механизми като TCP Wrappers или защитна стена.

Една от първите задачи на атакуващите е да установят версията на използвания софтуер на отсрещната страна, защото тя определя способите, които могат да бъдат използвани с напредване

на атаката. В тази връзка, една от най-популярните STO техники е скриване на версията на софтуера. При използване на уеб сървъри, версията на Nginx или Apache може да бъде скрита, което може да попречи на нападателите да разберат дали се използва уязвима и остаряла версия и да „настрои“ атаката спрямо нея.

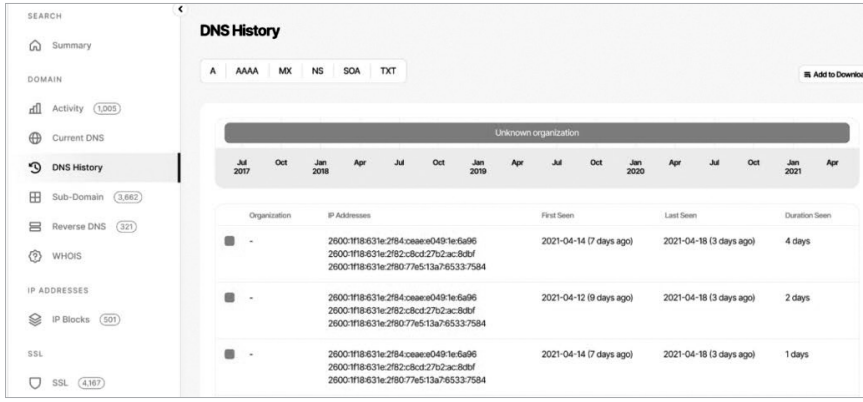
От представените тук примери и практики може да се направи погрешен първоначален извод, че STO е погрешен принцип. За да сме точни обаче, трябва да отбележим, че сигурността чрез неизвестност е добра, когато се комбинира с други механизми за сигурност и защитни правила. Но ако се разчита единствено на STO, вместо да се приложат практики за постигане на истинска сигурност на системата, всичко се губи веднага, щом нейните тайни бъдат разкрити. С други думи, няма нищо лошо в STO, има само лоши практикуващи.

Остава въпросът относно правилният начин за използване на STO. STO може да бъде много ефективен начин за намаляване на шансовете за атака, когато се използва заедно с други слоеве за сигурност, като ограничения, базирани на IP, 2FA, TCP Wrappers, защитна стена и други подобни. Силата на скриването не трябва да бъде подценявана, защото това е допълнителна мярка към цялата верига за сигурност.

STO е солидна практика, но скриването на информация винаги не винаги е добро, това зависи от типа на системата или приложението. Понякога това как и особено къде са скрити данни може да улесни нападателите при разкриване на тайните, така че предварителното планиране и оценка на ефекта е от съществена важност. Основният въпрос при вземането на крайно решение относно прилагането или не на STO е: ще помогне ли STO за намаляване на възможността или въздействието на входяща атака? Ако отговорът е да, няма причини да не се използва STO. Ако отговорът е не, значи трябва да бъдат намерени други възможности за защита.

Ако STO е част от защитната стратегия за киберсигурност, могат да бъдат използвани различни инструменти, като например SurfaceBrowser (https://securitytrails.com/corp/surfacebrowser), с които да бъдат одитирани всички онлайн активи, за да се открие критична разкрита информация, като отворени портове, DNS записи, IP адреси, SSL сертификати и много други. Един от възмож-

ните проблеми, които такъв вид инструменти може да открие е разкриването на DNS записи на вътрешни поддомейни по погрешка. Всички екипи за разработка създават тестови, dev, пробни или предполагаеми „вътрешни“ или „частни“ поддомейни по време на фазата на разработка. За съжаление, те често забравят да изтрият DNS записите, което води до масово създаване на остарели DNS записи. Остарелите DNS записи и дори други активни записи на поддомейни, станали публични по погрешка, се превръщат в често експлоатирана област. Приложенията за одит като SurfaceBrowser позволяват филтриране на поддомейни по ключови думи и изтегляте всички резултати за офлайн анализ (фиг. 17).



Фигура 17. Екран от приложение за одит

Сигурност чрез неизвестност е стара практика, която е полезна, когато се комбинира с други силни мерки за сигурност. Само по този начин трябва да се използва за минимизиране на въздействието на атака върху всяка организация. Да бъдеш изцяло зависим от него е твърде рисковано.

Друг, доста популярен инструмент за одит на сигурността на Word Press тема може да бъде открит и използван на адрес https://patchstack.com/auditing/ Той позволява анализ на добавките кода, базата данни с цел откриване на слабости в уеб приложението. Различни аналитични модели за анализ на сигурността и на резултатите от тестването също е подходящо да бъдат приложени (Koleva, 2022).

Четвърта глава

НАЦИОНАЛНИ ПОЛИТИКИ И ПРИНЦИПИ ЗА ПРИЛАГАНЕ НА СИГУРНОСТ ПО ДИЗАЙН ПРИ РАЗРАБОТКА НА СОФТУЕРНИ СИСТЕМИ

Някои от държавите, обект на голямо количество атаки и отделящи сериозни средства за опазване на сигурността са разработили собствени политики, модели и насоки за киберсигурност, включително сигурност на софтуерните приложения и услуги, базирани на принципа „Сигурност по дизайн“. Такива страни са Великобритания, Израел, Сингапур и др. Разбира се, на ниво Европейски съюз също са приети и разпространени такива политики и насоки, но до момента те са основно ограничени до на сигурността на личните данни и ще бъдат разгледани в съответния раздел. Тук ще разгледаме принципите, предлагани от съответните организации, отговорни за сигурността във Великобритания и отчасти в Сингапур, доколкото те са достъпни и могат да бъдат използвани и прилагани като методология и от организации от други страни, или елементи от тях да бъдат оценени като полезни и интегрирани в политиките за сигурност на организациите при синхронизиране на системите им за информационна сигурност с изискванията на стандартите.

Принципите на Обединеното Кралство за сигурност са разработени от Националния център по киберсигурност и на практика следват логиката на стандартите в софтуерното инженерство и са в категориите разработка и услуги (вериги за доставка). Те са достъпни и могат да бъдат намерени на адрес https://www.ncsc.gov.uk/collection/cyber-security-design-principles Експертите на Европейският институт за риск политики ERPI (European Ris Polisy Institute – risk-policy.eu/bg/) също препоръчват тези политики като основа за разработване на система са управление на сигурността в електронна среда.

Сигурните принципи на проектиране представляват набор от ръководства за проектиране на системи за киберсигурност. Тези принципи имат за цел да помогнат да се гарантира, че мрежите и

технологиите, които са в основата на съвременния живот, са проектирани и изградени сигурно.

За да бъдат полезни, системите много често трябва да трансферират, съхраняват и предоставят достъп до чувствителни данни. За съжаление, това ги прави основни цели за кибератаки. Ако тези системи бъдат успешно компрометирани, последствията могат да бъдат вредни, скъпи и неудобни. Често най-лошите резултати могат да бъдат избегнати, ако услугите са проектирани и управлявани със сигурност като основно изискване. Наборът от принципи, имат за цел да ръководят софтуерните инженери при създаването на системи, които са устойчиви на атаки, но и по-лесни за управление и актуализиране.

Като основа на принципите е използван системния дизайн при проектирането на софтуерни архитектури. Ръководството използва термина „система“, в смисъла на „колекция от цифрови компоненти, които са свързани чрез комуникационни технологии за изпълнение на бизнес функция.“. Един от основните обекти на ръководството са системите, част от електронното управление, която например онлайн услугата за кандидатстване за паспорт в Обединеното кралство, но то може да се отнася за много други цифрово базирани бизнес функции.

Друг, използван в ръководствата термин е „киберфизична система“, под който по дефиниция се разбира „система, която измерва или контролира физическия свят за постигане на определена цел“. Добър пример е модерен автомобил, в който сложна логика измерва физическата среда, за да контролира движението на автомобила, като тази дефиниция описва и повечето IoT системи и решения, т.е. ръководствата са приложими и по отношение на този вид системи.

Принципите са замислени да бъдат приложими както към цифрови системи, така и към киберфизически системи. Националният център обаче, обръща внимание на факта, че прилагането на принципите на практика за конкретни ситуации най-вероятно ще изисква известно персонализиране, за да отговаря на конкретните изисквания. Например, точните изисквания на онлайн информационна услуга ще бъдат различни от дистанционното управление

на електроцентрала. Въпреки това, принципите имат смисъла на ръководни и могат да бъдат приложени и в двата случая. Принципите за киберсигурност предлагат най-общо приложимите съвети. Принципите на проектиране на виртуализация се прилагат в по-специфичния случай на системи, които разчитат на технологии за виртуализация.

Наборът от принципи е разделен в пет категории, в зависимост от етапите, на които една атака може да бъде смекчена:

- Установяване на контекста преди проектирането – определяне на всички елементи, които съставят системата, така че защитни мерки, които ще бъдат разработени впоследствие да нямат слепи петна. За да е възможно създаването на проект за защита на системата, трябва да имаме добро разбиране на основите и да бъдат предприети действия за отстраняване на всички идентифицирани недостатъци.

- Затрудняване на атаката – нападателят може да се насочи само към частите на системата, до които е възможно да достигне, затова системата трябва да е възможно най-трудна за проникване. Проектирането с мисъл за сигурността означава прилагане на концепции и използване на техники, които затрудняват нападателите да компрометират данните или системите.

- Намаляване на ефекта от смущенията – създаване на система, която е устойчива на атаки за отказ на услуга и пикове на използване. Когато услуги с висока стойност или критични услуги разчитат на технология за доставка, става важно технологията да е винаги достъпна. В тези случаи приемливият процент за недостъпност на услугата може да бъде ефективно нула.

- Улесняване на откриването на пробив – системата трябва да бъде проектирана така, че възможно най-бързо да се забележи подозрителна дейност, когато се случи, и да може да бъдат предприети необходимите действия за нейното прекратяване. Дори и да бъдат взети всички налични предпазни мерки по отношение на известни уязвимости и атаки, все още има шанс системата да бъде компрометирана от нова или непозната атака. За да се осигури

най-добрия шанс за навременно откриване на тези атаки, трябва да са налични средства за постоянен мониторинг.

● Намаляване на въздействието от пробива – ако нападател успее да се интегрира в системата, той ще започне да я използва. Това трябва да е възможно най-трудно. Разработеният дизайн трябва естествено да минимизира тежестта на всеки пробив.

Ще разгледаме последователно принципите за всяка една от категориите. Всяка категория съдържа препоръчителни действия, които да бъдат извършени на този етап.

1. Установяване на контекста преди проектирането

1.1. Дефиниране на системата – трябва да разберем за какво служи системата, какво е необходимо за нейното функциониране и кои рискове са приемливи.

Важно е да се достигне до ниво на ясно разбиране за целта на всяка система. Трябва да бъдат идентифицирани и описани данните, връзките, хората и връзките с други системи ще са необходими, за да може нашата система работи и да изпълнява всички предвидени функции. Тук трябва да отбележим, че е желателно анализът да бъде направен по отношение на цялата система а не само за модулите, които предстои непосредствено да бъдат разработени, за да може да се разкрият всички зависимости и повторения по отношение на необходимите данни и потока на тяхното движение от мястото на генериране до мястото за съхранение и обработка.

Трябва да се определят въздействията, които се категоризират като неприемливи. Примерите могат да включват: неупълномощен достъп за преглед, модифициране или унищожаване на данни или недостъпност на системата за потребители за определен период от време, опити за извършване на измама и др. Добра практика е да бъдат разгледани примери от други, по възможност сродни по дейност организации, където нещата са се объркали, и да се оцени какво би означавало това в контекста на собствената

организация. Направените заключения би трябвало да са част от анализ на риска.

За да се приемат информирани дизайнерски решения, трябва да знаем кои рискове са приемливи. Рисковете, които организацията е готова да поеме трябва да бъдат документирани, утвърдени от мениджмънта на високо ниво и да сме уверени, че всички хора, участващи в проектирането на системата, са запознати с тях, така че да могат да вземат добре обосновани решения.

1.2 Да се разбере модела на заплаха за системата

На този етап се препоръчва използване на техники за моделиране на заплахи, като дървета на атаки (Shevchenko, 2018), за да ви помогнат да откриете начините, по които нападателят може да реализира целите си. Дизайнерът трябва, също така, да обмисли какво ниво на способност ще е необходимо на атакуващия, за да бъде успешен, и дали целта ви е защита, откриване или възстановяване, заедно с всякакви полезни, ограничени във времето цели. След като бъдат разбрани тези елементи, можете да се съпоставят контролите за сигурност към тези атаки, за да се проектира подходящо ниво на сигурността.

1.3 Да се разбере ролята на доставчиците при установяването и поддържането на сигурността на системата

Доставчиците, които компанията избирате за изграждането и експлоатацията на системата, играят жизненоважна роля в нейната защита. Важно е всички страни да разбират своите отговорности. Договорите с доставчици трябва да включват и изисквания за сигурност, които са достатъчно ясни, но прекаленото предписание може да доведе до противопоставяне и отказ от предоставяне на услугата. По-добре е да се разработи предложение за споделен риск с доставчиците, така че те да бъдат въвлечени в изпълнението на целите по сигурността, вместо просто да изпълняват договорно задължение. Насоките за сигурност на веригата за доставки на NCSC са предназначени да помогнат при установите на ефективен режим на контрол и надзор на доставчиците. (https://www.ncsc.gov.uk/ collection/supply-chain-security). Във всички случаи е добра

практика да се избират доставчици с вече сертифицирани системи за информационна сигурност.

1.4 Да се разбере системата „от край до край“

Трябва да бъдат разбрани критичните информационни и/или комуникационни потоци, на които система разчита (или ще разчита), за да работи. Трябва да бъде взета предвид всяка възможна точка, в която данните могат да бъдат съхранени, манипулирани или изобразени. Следните области често се пренебрегват:

● устройства, използвани за достъп до данни – ако данните се показват или обработват на устройство, трябва да се приеме, че данните присъстват на това устройство. Всички данни, до които потребителят има достъп, могат да бъдат достъпни за зловреден софтуер на устройството на потребителя;

● услуги на трети страни – изнесените доставчици на поддръжка, доставчиците на хостинг услуги и средите за управление на системните интегратори често остават извън обхвата, когато се разглежда сигурността на дадена система. Нападател с достъп до една от тези среди може да се опита да получи достъп до системата на фирмата;

● устройства за мрежова сигурност – проксита за сърфиране в мрежата и други устройства за наблюдение на мрежата, които обикновено се използват в корпоративни среди, може да дешифрират трафика между системата и нейните потребители. Тези устройства може да имат достъп до големи обеми чувствителни данни и да бъдат използвани от нападатели;

● копия на данните – трябва да се помисли за копия на данни, съхранени в журнали за проверка и инструменти за наблюдение, или копия, които са били експортирани в инструменти за бизнес разузнаване или информация за управление;

● комуникации през несигурни мрежи – ако система комуникира по канали, които не са физически защитени, то дизайнът ще трябва да включва съответните технически контроли, за да осигури подходящо ниво на поверителност и цялост;

● подходяща сигурност за всяка итерация на система – по време на процеса на проектиране можете да бъдат създадени отделни итерации на системата за различни цели, като разработка,

тестване и производство. Въздействието на компромиса, свързан с тези среди, вероятно ще варира в различните фази от жизнения цикъл на системата и трябва да бъде внимателно обмислено. Това е особено важно за сложни промишлени проекти и киберфизични системи, където голям брой различни компоненти са интегрирани, за да образуват цялостна система.

1.5 Да се изясни как се управляват рисковете за сигурността

Доброто управление предполага ефективен контрол върху сигурността на системите и операциите, а не сляпо придържане към предварително определени процеси. Когато дизайнерските решения изискват да се балансира между сигурност, използваемост и цена, важно е да се обърне внимание за щетите и въздействието върху бизнеса, вместо да се разчита на технически доводи. Трябва да се обърне внимание на мениджмънта, че цената да не бъде направено нещо понякога е точно толкова, а понякога и по-висока от цената да бъде направено. Разходи могат да включват глоби съгласно GDPR или законодателството за NIS, както и бизнес разходи и щети върху репутацията.

1.6 Да няма неясноти относно отговорностите

Всеки, който участва в проектирането и експлоатацията на система, трябва да има подходяща квалификация или опит, да знае каква е ролята му и да знае какви решения е упълномощен да взема. Правилните хора трябва да са упълномощени да защитават критични системи и понякога може да означава да се даде възможност на сравнително по-млади хора да влияят върху бизнес операциите. Това може да се разшири до умишлено намаляване на функционалността или нивата на обслужване в отговор на външни събития – без позоваване на висшето ръководство. Трябва да бъде въприет подход за непрекъснато развитие на уменията и обучението. Това ще гарантира, че пропуските в способности са идентифицирани, регистрирани и смекчени. Когато не е наличен подходящ експертен опит, свързаният риск трябва да бъде ескалиран и управляван като част от системата за управление на риска на организацията.

2. Компрометирането на системата трябва да е трудно

Проектирането с мисъл за сигурността означава прилагане на концепции и използване на техники, които затрудняват нападателите да компрометират данните или системите.

2.1 На външното въвеждане не може да се вярва. Данните трябва да бъдат валидирани и трансформирани преди да влязат в системата.

Всички данни от външен или по-малко надежден източник може да са били създадени, за да атакуват системата.

Добре структурираните данни могат да бъдат валидирани, за да се гарантира, че отговарят на очаквания формат. Валидирането на практика е проверка за това дали структурата и съдържанието на данните или файловете отговарят на очакванията, за да се гарантира, че те няма да инжектират злонамерен код в целевата система или да имат нежелани ефекти. Това е техника, на която може да се разчита само за сравнително прости файлови формати.

Ако валидацията не е възможна, те трябва да бъдат трансформирани. Има случаи, като например pdf файлове, когато е трудно или невъзможно да се направи проверка за зловреден софтуер. В тези случаи е най-добре да се трансформира файла или съдържанието в друг формат. Това трябва да има ефект на изрязване на всяко злонамерено съдържание, преди файлът да бъде предаден до местоназначението си.

Ако данните не могат да бъдат трансформирани, то те трябва да бъдат изобразени в среда, която няма проблем да бъде компрометирана, като например временна виртуална машина или отделна работна станция. Ако се импортира софтуер или двоични файлове, трябва да се направи проверка на криптографските подписи, за да сме сигурни, че софтуерът наистина е създаден от доставчик, на когото може да се има доверие.

2.2 Намаляване на повърхността за атака

Трябва да се предостави достъп и да се използват само интерфейсите, необходими за работата на системата. Ако бъдещите нападатели не могат да достигнат до интерфейс, те не могат

да го атакуват. Добра практика е да се премахнат всички конфигурации по подразбиране и функции, които не са необходими, като потребителски акаунти, пароли, скриптове и демонстрационни възможности. Когато се прави надграждане върху общи инструменти или софтуер, всички компоненти и библиотеки, от които не се използват трябва да бъдат деактивирани. Бизнес системата трябва да бъде разделена на зони за достъп и нива на сигурност. На външните нива потребителите трябва да имат права основно за четене, като обработката на данни се извършва във вътрешните, защитени зони. Ако нападател компрометира външната зона не би трябвало да може да въздейства на бизнес функционалността.

2.3 Увереност в ключовите контроли за сигурност

Трябва да се анализира и идентифицират компонентите на системата, които осигуряват най-важните контроли за сигурност и да сме уверени, че работят според очакванията. За тази цел трябва да бъде проверено:

- дали доставчиците на услуги са надеждни и компетентни в киберсигурността – за доставчиците може да се потърсят и проверят официални сертификати и одитни доклади;
- дали отделните продукти и услуги са добре проектирани, разработени и експлоатирани – за конкретна услуга или продукт можете да се провери реакцията на разработчика в отговор на открити уязвимости;
- дали внедреният продукт или услуга е добре конфигуриран и остава добре конфигуриран през целия му живот – за конкретни внедрявания тестът за проникване ще даде подробна обратна връзка.

Усилията трябва да бъдат съсредоточени върху контролите, които са най-важни за сигурността на системата.

2.4 Защита на средата за управление и операции от целенасочени атаки

Нападателите често се насочват към привилегировани потребители (администратори, инженери и т.н.) с фишинг имейли

или друг вид атака, базирана на социално инженерство. За да се избегне възможност нападател да получи привилегированите достъпи, които тези потребители притежават, трябва да се проектира системата така, че потребителите да не могат да преглеждат имейли или да сърфират в мрежата от същия акаунт или устройство, което използват за извършване на своите привилегировани действия.

В идеалния случай архитектурата за системно администриране трябва да е проектирана така, че да използва отделна инфраструктура за управление. Като минимум това означава използване на бастион хостове. Въпреки това, трябва да се има предвид, че подходът за бастионен хост не избягва злонамерения софтуер на устройството на потребителя да може да контролира сесията на администратор на отдалечен работен плот. Този риск може да бъде смекчен само чрез премахване на възможностите зловреден софтуер да получи достъп до устройството на администратора (като под администратор се разбира всеки потребител с привилегировани права). Това може да стане, например чрез предоставяне на отделни администраторски устройства или чрез гарантиране, че администраторските устройства трябва да имат достъп до имейл и мрежата чрез отдалечен работен плот или подобна технология. (Introduction to identity and access management – https://www.ncsc.gov.uk/guidance/introduction-identity-and-access-management)

Съществуват редица различни архитектурни модели, които могат да се използват за проектиране на административен подход за ИТ системи. Бастионният хост е компютър със специално предназначение в мрежа, специално проектиран и конфигуриран да издържа на атаки, наречен така по аналогия с военното укрепление. Компютърът обикновено хоства едно приложение или процес, например прокси сървър или балансиращо натоварване, а всички други услуги се премахват или ограничават, за да се намали заплахата за компютъра. Той е защитен по този начин главно поради местоположението и предназначението си, което е или от външната страна на защитна стена, или вътре в демилитаризирана зона (DMZ) и обикновено включва достъп от ненадеждни мрежи или компютри. Тези компютри също са оборудвани със специални

мрежови интерфейси, за да издържат на атаки с висока честотна лента през интернет.

Терминът обикновено се приписва на статия от 1990 г., обсъждаща защитни стени от Marcus J. Ranum, който дефинира бастионния хост като „система, идентифицирана от администратора на защитната стена като критична силна точка в мрежовата сигурност. Като цяло бастионните хостове ще имат известна степен на отделя се допълнително внимание на сигурността им, може да се подлагат на редовни одити и може да имат модифициран софтуер“. (Ranum, 2020). Той също така е описан като „всеки компютър, който е напълно изложен на атака, тъй като е от публичната страна на DMZ, незащитен от защитна стена или филтриращ рутер. Защитни стени и рутери, всичко, което осигурява сигурност за контрол на достъпа до периметъра, може да се счита за бастионни хостове. Други видове бастионни хостове могат да включват уеб, поща, DNS и FTP сървъри. Поради тяхната експозиция трябва да се положат много усилия в проектирането и конфигурирането на бастионни хостове, за да се сведат до минимум шансовете за проникване.“ (Krutz, 2003)

2.5 Предпочитание към изпитани подходи

Изграждането на нещо по поръчка, когато има различни опции за готови продукти, които можете да бъдат използвате, не е винаги за предпочитане. Това е особено важно при разработването на софтуер и при криптографията.

Популярните софтуерни рамки и библиотеки често са добре поддържани и тествани, с общност от разработчици, които активно търсят и поправят уязвимостите. Когато се надгражда върху тези готови решения се печели предимството на това тестване и проверка. Все пак трябва да извършват свои собствени проверки и оценки на всеки софтуер на трета страна.

Когато става въпрос за криптография, проектирането на нови техники е изключително трудно и трябва да бъде извършвано само от експерти по криптография. За предпочитане е използването на съществуващи алгоритми и протоколи.

2.6 Всички операции трябва да бъдат индивидуално разрешени и осчетоводени

Чувствителни или привилегировани действия трябва да бъдат разрешени само чрез функция за контрол на достъпа, която може да провери дали потребителят е този, за когото се представя, и че има разрешение за достъпа, който иска. Способността да се приписват действия на индивиди, а не на групи, е важна, когато става въпрос за установяване на отчетност. Това ще подпомогне и реакцията при инциденти. Ако имате непреодолимо изискване за споделяне на идентификационни данни, за да се позволи непрекъсната работа на система от контролна зала, могат да се използват физически и процедурни контроли за постигане на същия резултат. Това може да включва физически контрол на достъпа и видеонаблюдение.

2.7 Дизайн за лесна поддръжка

Лошо поддържаната система е уязвима. Трябва да се създаде механизъм за постоянно следене за съвети и корекции за сигурност и способност или за смекчаване на проблемите, или бързото им адресиране. Уязвимостите в сигурността трябва да бъдат отстранени незабавно или чрез софтуерни корекции, или чрез предприемане на други смекчаващи действия. Честите малки актуализации са за предпочитане пред редките големи, тъй като имат по-нисък рисков профил. Увеличаването на честотата на внедрявания също създава доверие в механизмите за внедряване и гарантира, че екипите са добре дисциплинирани при връщане назад на промените. Системата трябва да е проектирана така, че да не са необходими прекъсвания, които оказват влияние върху бизнеса, за прилагането на актуализации.

2.8 Администраторите трябва да разполагат с лесни инструменти за управление и контрола на достъпа

Наличието на унифициран изглед на потребители и разрешения за система или системи може да помогне на администраторите да поддържат контрол на достъпа по-лесно. Дизайнът трябва да поддържа процес на управление на жизнения цикъл на само-

личността (напр. за присъединили се и напуснали, хора, които променят ролите си и идентификационни данни за спешни случаи като възстановяване на парола, ако е необходимо). Възприемането на подход с единично влизане вместо дублиране на подобни системи за идентификация и контрол на достъпа може да опрости управлението на жизнения цикъл на самоличността.

2.9 Улеснявайте потребителите да постъпват правилно

Пробивите в сигурността често се случват, защото потребителите са разработили заобиколни решения за системни неадекватности. Трябва да се разгледат всички потенциални възможности потребителите да могат да заобикалят функциите за сигурност.

3. Намаляване на вероятността за смущения/атаки

Когато услуги с висока стойност или критични услуги разчитат на технология за доставка, става важно технологията да е винаги достъпна. В тези случаи приемливият процент на престой може да бъде ефективно нула.

3.1 Системите трябва да са устойчиви както на атака, така и на отказ

За да се отстрани проблема, обичайна практика е да се осигурят резервни системи, алтернативни маршрути и архивиране на данни. Те се представят добре срещу случаен отказ или грешки, но често имат по-малък ефект срещу злонамерени атаки.

Например, ако системата разполага с 10 идентични сървъра с балансирано натоварване и всеки има шанс 1 към 10 за произволен отказ, шансовете всички те да се повредят наведнъж са 1 към 10 000 000 000. Въпреки това, ако всички те имат една и съща уязвимост, за атакуващия не е проблем да атакува всички 10, а не само един.

Вторият пример касае резервното копие. Резервните копия на данните са задължителни за работата на системата. Въпреки това, ако атакуващият може лесно да изтрие или повреди резервните копия, те ще бъдат полезни само за възстановяване от случайни повреди и грешки.

И накрая, за киберфизическите системи се прилагат контроли за безопасност, за да се намали рискът от опасен резултат. Този процес трябва внимателно да разгледа възможността външен участник да промени целостта или наличността на контролите за безопасност.

Архитектурите на системите за безопасност трябва да бъдат проектирани така, че да гарантират, че неприемливите последствия са твърде скъпи за нападателя. За да се получи този резултат, контролите за безопасност трябва да бъдат независими както в случай на компрометиране на системата, така и на механична повреда.

3.2 Дизайн, гарантиращ мащабируемост на системата

За да може системата да се справи с изключителни пикове в търсенето или бъдещо разширяване, може да се наложи бързо мащабиране услугата. Отчитането на потенциала за бъдещо търсене на ранен етап трябва да означава, че системите са по-лесни за мащабиране по-късно. Те, също така трябва да са по-подходящи за мащабиране при повишено търсене или когато са атакувани.

3.3 Да се идентифицират тесните места, възможности за високо натоварване и условия на отказ от обслужване

Трябва да се идентифицират всички тесни места в системата. Например нисък капацитет, наследена бизнес технология или основна микроуслуга, която се обажда на услуга на трета страна. Уверете се, че имате план за справяне с тези затруднения по време на периоди на високо натоварване или прекъсване. При създаване на цялостната стратегия и план за тестване на системата трябва да се добавят специфични тестове за необичайно високо натоварване и за отказ на услуга. Например, можете да се симулират някои атаки за отказ на услуга, като целенасочено се прекратят определени микроуслуги или инфраструктурни елементи в предпроизводствените среди.

Съществуват и свободно достъпни инструменти, като например Chaos Monkey на Netflix (https://github.com/Netflix / SimianArmy/wiki/Chaos-Monkey), които могат да бъдат използвани за тестване на поведението на системата при високо натоварване

или когато компонентите се повредят. Важно е да се тества реакцията на условия на отказ, както и да се анализира какви биха могли да бъдат тези условия на отказ.

3.4 Трябва да се идентифицират местата, в които наличността зависи от трета страна и да се планира провала на тази трета страна

Много организации разчитат на услуги на трети страни, като телекомуникационни връзки, хостинг, удостоверяване или услуги за системно администриране. Трябва да разбираме характеристиките на наличността на тези услуги на трети страни и въздействието върху операциите, ако се провалят, особено в моменти на критично търсене. Трябва да има разработен план за минимизиране на смущенията, ако възникне такова събитие.

4. Лесно откриване на пробиви

Дори и при всички налични предпазни мерки, винаги има шанс системата да бъде компрометирана от нова или непозната атака. За да се осигури най-добрия шанс за идентифициране на тази атаки, трябва системата да е в състояние да открие компрометиране.

4.1 Всички събития и регистрационни файлове за сигурността трябва да бъдат събирани и анализирани

Наличието на правилните данни е от съществено значение. Това е вярно, независимо дали целта е анализ в случай на пробив или откриване на потенциални и действителни компромиси в реално време. На етапа на проектиране трябва да сме сигурни, че системата регистрира достатъчно данни, за да се извърши анализ на основната причина в случай на повреда. Въпросът е дали регистрационните файлове ще съдържат необходимите данни, за да се разбере дали е възникнал отказ в резултат на пробив. Може да са необходими регистрационни файлове, както на ниво инфраструктура, така и на ниво приложение. Освен събирането на регистрационни файлове за съответните събития, трябва да се гарантира, че целостта на тези файлове ще бъде запазена в случай на пробив. Нападателят не трябва да може да прикрие следите си.

4.2 Проектиране на прости комуникационни потоци между компонентите

Добре обмисленият дизайн с ясно дефинирана и строго ограничена комуникация между компонентите може да опрости анализа на сигурността и да позволи да идентифициране на нередности. Компонентите, които се опитват да комуникират по начини, които не са част от дизайна, могат да бъдат ясна индикация за пробив. Инструментите за наблюдение трябва да са конфигурирани така, че да откриват тези индикатори и автоматично да изпращат предупреждения.

4.3 Откриване на командни и контролни комуникации на зловреден софтуер

Трябва да се следи за опити на компрометирани компоненти да се свържат с тяхната инфраструктура за управление и контрол. Това може да се постигне чрез разрешаване на изброяване на външни домейни или адреси, които са приемливи за изход на данни. Опитите за достигане до други домейни трябва да бъдат предотвратявани и преглеждани.

4.4 Наблюдението да е независимо от наблюдаваната система

Това гарантира, че ако наблюдаваната система е компрометирана, нападателят няма да има видимост дали пробивът е бил открит.

Същият принцип важи и за кибер-физични системи за контрол (като индустриални системи за контрол), където телеметрията и контролните канали трябва да се поддържат независими, ако е критично да се знае как се държи системата, независимо от действията на нападателя.

4.5 Атакуващите трябва да са затруднени при откриването на правила за сигурност чрез външно тестване

Ако активността на потребителя доведе до задействане на някое от правилата за сигурност, системата трябва да дава само минималната необходима обратна връзка на потребителя. Това

прави по-трудно за нападателя да разбере логиката за сигурност, когато се опитва да разгадае методите за защита.

4.6 Разбиране на „нормалното“ и откриване на ненормалното

Доброто познаване на нормалната работа на системите означава, че неочакваното поведение може да бъде по-лесно разпознато. В допълнение към проектирането на прости комуникационни потоци между компонентите, може да бъде полезно да се наблюдава натоварването на мрежата, I/O за съхранение, изчислителна производителност или транзакционна активност, за да се разбере кога системата се държи необичайно, което от своя страна може да е индикация за опити за атака.

5. Намаляване на въздействието на компромиса

Целта е да се разработи дизайн, който естествено минимизира тежестта на всеки компромис.

5.1 Използване на зониран или сегментиран мрежов подход

Сегментирането на активи в мрежа осигурява следните предимства:

- Помага да се ограничи компромисът в сегмента, който е бил нарушен.
- Позволява да се защитят по-добре активите, които са най-чувствителни или ценни.
- Поддържа възможността за ограничаване или изследване на комуникационните потоци между сегментите. Това означава, че могат да бъдат създадени правила за наблюдение, които са в състояние да твърдят с голяма увереност, че е настъпил пробив или неправилна конфигурация.
- Решенията за това как да се сегментира мрежа обикновено се вземат предвид защитите, които различните активи изискват, тяхната необходимост от взаимодействие с други активи и степента, до която се вярва на тяхната цялост.

5.2 Премахване на ненужната функционалност, особено когато неоторизираната употреба би била вредна

Ако съществува функционалност за оторизирани потребители, тогава тя може да бъде използвана от неоторизирани потребители в случай на компрометиране. Намаляването на наличието на ненужна функционалност ще намали този риск. По този начин ще се намалят и оперативните разходи за поддръжка на софтуер или функционалност, която не е необходима, опростявайки системата и улеснявайки наблюдението. Премахването на ненужната функционалност може да приеме няколко форми, като по-точни настройка и конфигурациите на софтуера, забрана на функции за автоматично изчистване на грешки и др.

5.3 Не трябва да се създава ситуация от вида „байпас за управление“

Често срещан недостатък на дизайна е да има по-слаби контроли за сигурност и архитектура за сигурност в управленските комуникации, отколкото в управляваните системи. В такива сценарии компрометирането на единичен външен компонент може да доведе до привилегирован достъп до системи или данни през канали, които са предназначени само за административна употреба.

5.4 Улеснено възстановяване след компромис

Системната архитектура трябва да се проектира по начин, който позволява при откриване на компромис бързо възстановяване до познато чисто състояние, след като недостатъкът, довел до компромис е бил адресиран.

Създаденият дизайн трябва да позволява, както възстановяване, така и поддържане на записите и данните, които може да са необходими в подкрепа на разследване на инцидента. При някои инциденти може да се окаже, че трябва да се направи избор между бързо възстановяване на системата или запазване на данните, нужни за разследване. Това може да стане и ако не разполагаме с достатъчно инфраструктура за дублиране на системата. Проблемът трябва да е обект на оценката на риска.

5.5 Дизайн в подкрепа на „разделяне на задълженията“

Когато въздействието от атака, злоупотреба или компрометиране би било значително, може да се обсъди изискване най-привилегированите или потенциално опасни функции в системата да изискват двама или повече лица да работят заедно, за да ги изпълняват.

Пример: как може да се попречи на един администратор (или негов акаунт) да експортира копие на всички данни или да извърши промени с голямо въздействие.

5.6 Анонимизиране на данни, когато се експортират в инструменти за отчитане

Инструментите за ефективност или отчитане трябва да се доставят с десенсибилизирани данни.

Да предположим, че има оперативен екип, който иска да създаде и покаже табло за управление на ефективността за система за финансови плащания. За да функционира правилно, тяхното табло за управление не трябва да работи върху необработени данни за транзакции, които може да са чувствителни. Всяка лична информация може да бъде премахната от данните, преди аналитичната система да я обработи. Това ще намали броя на местата, където може да възникне пробив със силно въздействие.

Препоръчително е прилагането на контроли за анонимизиране на данните възможно най-близо до източника. Вместо да се разчита на инструменти за статистика за анонимизиране на данните, трябва да се поддържа собствен контрол върху процеса.

5.7 Да не се позволяват произволни заявки към данните

Не е желателно да се проектират функционалности и да се внедряват приложения, които позволяват произволни заявки към базите данни. Тези приложения подкопават дизайна на сегментирани системи, като предоставят по-лесен път за компрометиране на данните.

5.8 Да се избягват ненужните кеширания на данни

Тези временни хранилища на данни обикновено са по-малко защитени от основното хранилище на данни, но потенциално могат да дадат информация с висока стойност на нападател.

Ако се изисква кеширане или временно хранилище на данни, то трябва да се управлява от политика за избледняване на данни, която изчиства записите възможно най-скоро след приключване на достъпа, като гарантира, че в кеша се съхраняват минимални данни и това, което се съхранява, е подходящо защитено.

6. Принципи на проектиране на сигурността при виртуализация

Тези принципи се фокусират върху технологиите за виртуализация, които могат да се използват в облака, на сървъри, разположени на място, или на устройства на крайни потребители. Принципите са подмножество от принципите за проектиране на киберсигурност, като ги разширяват, за да помогнат за осмисляне на съображенията за сигурност, при проектирате системи, които използват виртуализация.

Внедряването и поддръжката на дискретни системи, използващи физическа инфраструктура, като сървъри и мрежови рутери, може да бъде скъпо, времеемко и неефективно. Виртуализацията позволява на множество системи да споделят физически ресурси, позволявайки създаването и внедряването на много виртуални системи евтино и бързо. Тя е ценен инструмент за консолидиране на ресурси, поддръжка на наследени системи, увеличаване на гъвкавостта и намаляване на разходите. Въпреки, че често се използва за изграждане на изчислителни ресурси, тя може да се прилага към различни слоеве на системата, включително мрежи и съхранение (Kolomoitcev, 2020). Тази технология е мощна, но също така въвежда допълнителни нива на сложност и, потенциално, допълнителни рискове в системите. Ето защо е важно да се гарантира, че сигурността се разглежда през целия процес на проектиране на всяка система, която разчита на виртуализация.

Тези принципи, дефинирани от отговорните за киберсигурността институции на Великобритания, имат за цел да помогнат и насочват вниманието за проектирането на системи, които се възползват от виртуализацията, без да въвеждат неуправляеми рискове.

6.1.Компоненти на виртуалната система

Съществуват много различни видове виртуализация, включително изчисления, работа в мрежа, съхранение, като непрекъснато се създават нови приложения. Акцентът в случая е върху виртуализацията на инфраструктурата, но тези принципи могат да бъдат приложени към всеки тип виртуализация.

Виртуалната система е система, изградена с помощта на множество виртуални екземпляри, работещи на платформата за виртуализация.

Виртуален екземпляр, това е логическият виртуален екземпляр, който работи на платформата за виртуализация. Може да е пълноценна виртуална машина, виртуална защитна стена или друго мрежово устройство, или виртуална файлова система. Виртуалният екземпляр обикновено е логическо представяне на физически ресурси, но може да бъде и допълнителен слой на абстракция, използван за привеждане на множество ресурси в един логически изглед.

Основният доставчик на виртуализация е платформата за визуализация. Платформата позволява множество, сегментирани виртуални инстанции да се изпълняват на нея и управлява ресурсите, които са им достъпни. Платформата за виртуализация може да включва много компоненти като хипервизор, оркестрация и функции за управление.

Платформата за виртуализация работи върху физическия хардуер. Често хардуерът има специфични функции, които позволяват ефективна виртуализация, така че хардуерът да може да се споделя между виртуални инстанции с почти естествена скорост.

Облачните услуги разчитат в голяма степен на виртуализация, за да им помогнат да предоставят услугите си. Принципите

за сигурност в облака (https://www.ncsc.gov.uk/collection/cloud – Cloud security guidance) могат да помогнат за изграждане на доверие в тяхната сигурност (Awaysheh, 2021).

6.2. Структура на принципите за сигурност при виртуализация

Тези принципи са разработени на база на рисковете, свързани с виртуализацията и дават насоки за смекчаване на тези рискове. Полезността на тези принципи е когато се използват при избора на изисквания по отношение на сигурността в началото на процеса на проектиране на решението. Принципите са обособени в пет раздела, всеки от които се занимава с конкретен аспект на сигурността на системата. Те следват логиката и в този смисъл частично повтарят принципите при проектиране и разработка на софтуерна система.

Първи принцип. Установяване на контекста

- Разбиране на рисковите профили на всички системи, споделящи платформа за виртуализация

Докато виртуализацията има за цел да осигури разделяне между виртуални инстанции, тази защита не е перфектна. Има уязвимости за избягване на виртуални машини, които позволяват на един виртуален екземпляр да взаимодейства с друг или с основната платформа за виртуализация. Има също така примери за уязвимости, които могат да причинят отказ от услуга на платформата за виртуализация, като по този начин засягат всички виртуализирани екземпляри на платформата.

Като се има предвид това, може да не е подходящо да се хостват виртуални инстанции с много различни рискови профили и въздействия от компрометиране на една и съща платформа за виртуализация. Например, виртуализирана система, работеща с киберфизична система, като индустриален контролен процес, заедно със система, свързана директно към интернет, хостваща уеб сървър. Ако има изискване за споделяне на платформа за виртуализация по този начин ще са необходими допълнителни контроли за управление на допълнителните рискове. Пример за такава контрола би било деактивирането на комуникационните канали на

виртуалния екземпляр към платформата за виртуализация, за да се намали повърхността на атака, но това също ще деактивира функционалността, съответно трябва да е обект на оценка.

- Когато виртуализацията не се използва като бариера за сигурност не са необходими допълнителни контроли за сигурност

Ако виртуализацията се използва за хостване на системи със същия рисков профил и въздействие на компромис може да се използва пълната и богата функционалност на платформата за виртуализация. Такъв е случаят, когато виртуализацията се използва за консолидиране на ресурси или подобряване на процеса по администриране.

- Разбиране за ролята на доставчиците при установяването и поддържането на сигурността на системата

Това е особено важно при изграждането на услуги върху платформи за виртуализация, предоставени от външни страни, като облачни услуги. Виртуализацията може да присъства в множество компоненти под различни форми, от изчисление до съхранение и работа в мрежа. Възможно е в първоначалното описание да няма информация за това, че част от компонентите са виртуализирани или че споделят ресурси с множество потребители.

Един пример за сценарий за споделяне на ресурси е софтуерно дефинирана мрежа, често разгръщана в центрове за данни. За крайните потребители изглежда, че има частна мрежа, като всъщност те я споделят с множество наематели на един и същ център за данни.

Трябва да се комуникира с доставчика, за да се разбере каква е използваната от него технология и дали е подходяща за конкретната система. В тези случаи е приложимо и ръководството за управление на веригите за доставка.

- Виртуализацията не смекчава уязвимостите в наследени и остарели системи

Виртуализацията може да бъде полезна за хостване на наследени и остарели системи, въпреки това, ако тези системи нямат функции за сигурност или имат уязвимости, това няма да се промени чрез виртуализирането им. Тези слабости все още могат да

бъдат използвани от нападател. Виртуализацията на остарели системи може да помогне за отделянето им от останалата част на инфраструктура та и сегментацията, но компрометирана виртуална инстанция може да се използва за стартиране на атака срещу други виртуални инстанции или основната платформа за виртуализация.

Втори принцип. Компрометирането на системата трябва да е трудно

Проектирането трябва да се извърши с мисъл за сигурността на всеки слой на виртуализираната система.

● Виртуализираните системи трябва да са защитени по същия начин, както и невиртуализираните

Платформата за виртуализация по своята същност не защитава виртуалните екземпляри, работещи на нея, от компрометиране. Целта на платформата е да позволи на множество виртуални компоненти да работят едновременно и да осигури разделяне между тях. Следователно, към виртуализирана система трябва да се приложат съответните контроли за сигурност, точно както и при невиртуализирани системи. Това ще спомогне за предотвратяване на компрометиране на целевия виртуален екземпляр и последващо компрометиране на други виртуални екземпляри на същата платформа.

● Непрекъснато актуализиране на виртуалните екземпляри и платформите за виртуализация

Виртуализацията на система без корекции не намалява риска тя да бъде компрометирана. Следователно актуализациите трябва да се прилагат към софтуера на виртуалния екземпляр, когато станат налични. От друга страна, виртуализацията може да се използва за рационализиране на методологията за актуализиране. Например, виртуална референтна система може да се стартира за тестване на актуализации, преди да се приложи към производствена система. Могат да бъдат направени моментни снимки на системата, преди да се приложи корекция, което дава възможност за връщане на промените назад, ако е необходимо. Това е особено важно при сложни процеси на миграция, които

могат да доведат до загуба на голямо количество данни, отказ от услуга и загуба на престиж.

Дизайнът на системата трябва да гарантира, че всеки компонент може да бъде лесно актуализиран. Използването на функции за виртуализация, като автоматично прехвърляне при отказ може да помогне за осигуряване на гъвкавостта. Внимателно трябва да се обмисли как платформата за виртуализация може да се актуализира и поддържа, без да се засягат виртуалните инстанции, работещи на нея. Актуализирането на основната платформа не трябва да оказва влияние върху операциите или потребителите.

- Защита на интерфейсите за управление на платформата за виртуализация

Нападател с достъп до административната конзола или инфраструктура за управление може да контролира конфигурацията на платформата за виртуализация и всички виртуални инстанции, работещи на нея. Само ограничен брой хора със специфични задачи трябва да имат достъп до административната конзола. Административната конзола трябва да е достатъчно защитена, чрез контроли като мрежова сегрегация, политика за силна парола, многофакторно удостоверяване и ролеви контрол на достъпа. Административните интерфейси с достъп до мрежата трябва да приемат само връзки от оторизирана инфраструктура за управление, която е отделена от нормалната системна инфраструктура.

- Защита на комуникационните канали между виртуалните инстанции и виртуалната платформа

Много виртуални платформи предоставят канали за комуникация между виртуални инстанции и към самата виртуална платформа. Функционалността на виртуалната платформа често се активира чрез тези комуникационни канали и затова е желателно да се използват. Важно е разбирането на последиците от използването на тези пътища за комуникация и баланса между риска и ползата, което трябва да стане на база на приемливия за организацията риск за конкретните операции. Тези интерфейси предоставят маршрут до платформата за виртуализация, която противникът потенциално може да използва, за да се придвижи хоризонтално между инстан-

ции или вертикално към платформата за виртуализация. В случай, че се използват, те трябва да се конфигурират според указанията за най-добри практики, предоставени от документацията на виртуалната платформа.

Трети принцип. Затрудняване на смущенията

Виртуализацията има редица функции, които могат да се използват, за да се избегнат смущения и да се подобри достъпността. Въпреки това, ако една система не е проектирана правилно, виртуализацията може да се превърне в единична точка на повреда и да намали устойчивостта.

- Проектиране на системата за виртуализация с излишък

Системите, които използват виртуализация, трябва да бъдат проектирани с оглед на излишъка. Атака за отказ на услуга на платформа за виртуализация може да повлияе на всички виртуализирани машини, работещи на нея. Това може да се постигне, например, чрез компрометиране на лошо защитена виртуална машина и използване на това за стартиране на атака срещу основната платформа. Приемливо ниво на прекъсване може да бъде проектирано, като се използват множество нива на резервиране, ако е необходимо. Мерките, които могат да помогнат с това, включват множество копия на хардуер, мрежови пътища, масиви за съхранение, физическо разпределяне на системата на различни местоположения, онлайн и офлайн архивиране.

- Дизайн, който да е гъвкав, мащабируем и високодостъпен

Много платформи за виртуализация имат функции, които им позволяват да бъдат гъвкави, мащабируеми и високо достъпни. Конкретният дизайн трябва да се възползва от тези характеристики, като допринася за създаването на стабилни и устойчиви системи. Използването на тези функции трябва да се прилага в целия стек за виртуализация. Това ще улесни и текущата поддръжка, като например прилагане на корекции за сигурност към основната платформа за виртуализация и инфраструктура, без да се засягат виртуализираните системи, работещи на тях.

● Инфраструктурата за управление да е високо достъпна

Ако платформата за виртуализация бъде прекъсната, инфраструктурата за управление трябва да е достъпна за администраторите, за да коригират проблема. Инфраструктурата за управление трябва да бъде отделна от оперативната инфраструктура, така че атаката срещу операциите да не може да засегне дейностите по управление.

Четвърти принцип. Улеснено откриване на компромиси

Дизайнът трябва да включва възможност за откриване на атаки и компромиси. Това ще позволи бърза реакция, както на опити, така и на успешни атаки. За целта е необходимо да бъдат въведени допълнителни контроли за сигурност, където е необходимо, като част от текущия жизнен цикъл на системата. Освен това тези контроли трябва редовно да бъдат проследявани а при необходимост и периодично променяни, за да не са обект на сканиране.

● Добро познаване на виртуалната инфраструктура

Система трябва да се проектира така, че да е лесна за одит. Трябва във всеки един момент да е възможно да се даде отговор на въпроса „Какво се изпълнява, къде и кои системи имат достъп?“. Неконтролираното използване на виртуализация може бързо да излезе извън контрол, което води до ситуация, в която има малък или никакъв надзор върху системите, работещи на виртуалната платформа. Работен процес за разработка трябва да включва процес за настройка и документиране на нови виртуализирани системи. Това е особено важно при внедряването на производствени системи, които са критични за организацията.

● Използване на функциите за виртуализация, за подобряване на одита и мониторинга

Във физическа среда проследяването на оборудването и „ИТ в сянка“ може да бъде трудно. Използването на виртуализация може да усложни този проблем, като позволи бързо и лесно създаване на виртуални системи, които може да не са защитени правилно или просто да бъдат забравени. Това е особено вярно в циклите на разработка на бързи версии при гъвкавите методоло-

гии, където фокусът е върху следващата доставка, а не върху изчистването на последната итерация. Виртуализацията всъщност може да улесни одита и мониторинга. Тъй като платформите за виртуализация се управляват от централна точка, е възможно да се проектира инфраструктурата за управление, така че да може да прави запитвания към виртуализираната система на всеки слой от стека. Гъвкавия характер на виртуализацията може да се използва, за да се добавят възможности за наблюдение във всеки слой на платформата и докладване обратно на едно централно място. Отлично решение в случая са и професионалните системи за управление на събития и инциденти, които са в състояние да събират информация от всички хоризонтални и вертикални нива на виртуализираната система.

Пети принцип. Намаляване на въздействието на компромиса

Функции като репликация, моментни снимки и висока наличност могат да се използват за ускоряване на възстановяването на системата, давайки на виртуалните системи предимство пред традиционната инфраструктура. Проектът на системата трябва да се възползва от тези силни страни, когато е възможно.

- Дизайн за бързо възстановяване

Виртуализацията предоставя функционалност, която да помогне за бързо възстановяване от компромис. Например, ако платформата за виртуализация се състои от множество възли, за да се създаде клъстер, виртуализираните системи могат да бъдат копирани в множество възли, осигурявайки опция за бързо възстановяване ако възел на инфраструктурата стане недостъпен.

Наличието на резервно копие на система в известно последно добро състояние, също може да бъде от полза при възстановяване от компрометиране. Това може да се постигне с помощта на моментни снимки или отделни референтни платформи. Трябва да сме сигурни, че тези резервни копия представляват най-новата итерация на системата и да се създадат процедури по непрекъснато обновяване на резервните копия.

- „Елегантно“ влошаване на услугата

Проектираната система трябва да имат възможност за леко влошаване на функционалността, предоставяйки минимален набор от услуги, в случай на инцидент, който пречи да бъде поддържана пълна функционалност. В ситуация, в която пълният набор от функции на системата не може да бъде доставен, може би поради кибератака или дори природно бедствие, гъвкавостта на виртуализацията може да бъде използвана, за да се реагира бързо на всяко влошаване на услугата. Например, виртуализацията може да осигури средство за бързо възстановяване на системи в добре познато състояние или мрежата може бързо да бъде преконфигурирана, за да насочва потребителите към система за преодоляване на отказ.

ЗАКЛЮЧЕНИЕ

Стандартите и принципите полагат само основата на информационната сигурност. Променящите се условия, развитието на технологиите по отношение на изчислителната мощност, алгоритмите за изкуствен интелект (Andeev, 2020), интензивния процес на дигитализация във всички области на икономическия и обществен живот (Andreev, 2021) поставят атакуващите във все по-изгодна позиция, като правят задачата за осигуряване на сигурността на информационните системи и данните все по-сложна. Тя трябва да присъства във всички фази на жизнения цикъл и до края на логическия живот на софтуера и да стане неразделна част от стандартните методологии за проектиране, разработка, тестване, поддръжка и експлоатация на софтуера.

ИЗПОЛЗВАНА ЛИТЕРАТУРА:

1. Андреев, Е., М. Николова, 2021. Анализ на функционалности и уязвимости на web-базирани информационни системи, *Съвременни изследвания и технологии за отбраната – ARTDef 2021, Институт по отбрана „Проф. Цветан Лазаров“*, II-17 – II-24. ISSN 2815-2581.
2. Арнаудов, Д., А. Иванова, Сигурност и защита на информационните системи, ВСУ „Черноризец Храбър“ – Университетски издателство, Варна, 2007
3. Илиев, Н., С., Разработка на система за планиране на парични потоци., В: Дипломна работа – Варненски свободен университет, Варна, 2021, с.70.
4. Кръстев Др., Правни основи на киберсигурността. Том 1: Теоретични и международно правни аспекти, ИК Стено, Варна, 2021
5. Кръстев Др., Правни основи на киберсигурността. Том 2: Технологични и наказателно правни аспекти, ИК Стено, Варна, 2021
6. Национална стратегия за киберсигурност „Киберустойчива България 2023“, София, 2021
7. Пенева Ю., Бази от данни, първа част, изд. Регалия 6, София, 2004
8. ПЕТРОВА, В., Модел за многокритериално вземане на решения, за осигуряване на киберсигурност на софтуерни приложения. Научно списание „Известия“ на съюза на учените – Варна, секция „Технически науки“-1, 2021, ISSN1310-5833.
9. Сребров П., Защита на информацията при работа с ЕИМ, Държавно издателство „Техника“, София, 1989
10. Andreev, E., M. Nikolova, V. Radeva, Educational NASA Project: Artificial Intelligence and Cybersecurity at a Mobile Lunar Base. *Information & Security: An International Journal* **46**(3), 2020, 321 – 333, https://doi.org/10.11610/isij.4624

11. Andreev, E., V. Radeva, M. Nikolova, 2021, Cybersecurity of information in space telemedicine, Proceedings of 15th International Conference of Communications, electromagnetic and medical applications, Faculty of telecommunications Technical University of Athens, Greece, 54 – 57. ISSN: 1314-2100.

12. Agboola, R. B., Iro, Z. S., Awwalu, J. ,Said, I. N., Database Security Framework Design Using Tokenization. In:// DUJOPAS, vol.8 №1b pp.16-26, March 2022, DOI:10.4314/dujopas.v8.i1b.3,Aviable from: https://www researchgate. net/publication 360536197].

13. Agrawal, P., Singh, A., Raghavan, M., Sharma, S., and Banerjee, S. An operational architecture for privacy-by-design in public service application., In: //arXiv:2006.04654v1[cs.CR] 8 Jun 2020.

14. Al-Matouq, H., Mahmood, S., Alshayeb, M., Niazi, M. A Maturity Model for Secure Software Design: A Multivocal Study. // IEEE *ACCESS*, vol 8,2020, DOI:10.1109/*ACCESS*.2020.3040220., https://creativecommons.org/licenses/by/4.0/.

15. Awaysheh, F.M., Aladwan, M., Alawadi, S., Pena, T.F.,Cabaleiro, J.C., Security by Design for Big Data Frameworks Over Cloud Computing. In:// IEEE Transactions on Engineering Management, February 2021, Available from: https://www researchgate. net/publication 349148582].

16. Barbosa, M., Mokhtar, S.,Felber, P.,Maia, F.,Matos, M., Oliveira,R., Riviere, E., Schiavoni, V., Voulgaris, S., SAFETHINGS: Data Security by Design in the IoT. //Conference Paper. September 2017, DOI:10.1109/EDCC.2017.33, Aviable from: https://www researchgate. net/publication 321400928].

17. Boldea, C., Krz. Socha, Power Shell – Cybersecurity Perspective, SERT-EU Security Whitepaper 2019 – 001,

18. Bu, F., Wang, N., Jang, B., and Liang, H.,Privacy by Design., implementation: Information system engineers perspective., Available from: https://doi.org/10.1016/j.ijinfomgt.2020.102124. International Journal of Information Management 53(2020)102124.

19. Cavoukian, A., Privacy by Design., In://IEEE Technology and Society Magazine, 2012, p.19., DOI:10.1109/MTS.2012.2225459.

20. Cavoukian, A., The 7 Foudational Principles. Implementation and Mapping of Fair Information Practices. Cavoukian, Ann, Information &Privacy Commissioner, Ontario, Canada. //Creation of a Global Privacy Standard (November 2006), at ://www.ipc.on.ca/images/Resources/ gpc.pdf.

21. Cavoukian, A., Information & Privacy Commissioner, Ontario, Canada. Creation of a Global Privacy Standard, November 2006, Aviable from [https://www.privacysecurityacademy.com/wp-content/uploads/2020/08/PbD-Principles-and-Mapping.pdf]

22. Cavoukian, A., Understanding How to Implement Privacy by Design, One step at a Time., In://IEEE Consumer Electronics Magazine, 2019. DOI:10.1109/MCE.2019.2953739.

23. Casola, V., De Benedictis, A., Rak, M., Villano, U. A novel Security-by-Design methodology: Modeling and assessing secuhity by SLAs with a quantitative approach.//The journal of Sistems and Software 163(2020)110537., https://doi.org/10.1016/j.jss.2020.110537.

24. Colesky, M., Hoepman, J., Hillen, Ch., A Critical Analysis of Privacy Design Strategies. In: // Conference Paper. May 2016. DOI:10.1109/SPW.2016.23. Aviable from: https://www researchgate. net/publication 305870977].

25. Dietz, M., Hornuny, C., Hagemann, L., Pernul, G., Employing Digital Twins for Security-by-Design System Testing. // Conference Paper. April 2022, DOI:10.1145/3510547.3517929, Available from: https://www researchgate. net/publication 360312462].

26. Dantas, Y. G., Nidam, V., Automating Safety and Security Co-Desing through Semantically-Rich Architecture Patterns. In://ACM Transactions on Cyber-Physical Systems, vol. 5, №. 3, Article 111, July 2021, Available from: https://www researchgate. net/publication 358142955].

27. European Commission Directorate-General for Communication Security standards applying to all European Commission informa-

tion systems. EU ITSRM, IT SECURITY RISK MANAGEMENT METHODOLOGY V1.2. [Online] Available at: [https://ec.europa.eu/info/publications/security-standards-applying-all-european–commission-information-systems_en]

28. Ettinger, J., Cyber Intelligence Tradecraft Report, The State of Cyber Intelligence Practicec in the Unated States, Carnegy Melon University, 2019, Avialble from [https://resources.sei.cmu.edu/asset _files/EducationalMaterial/2019_011_001_546699.pdf]
29. Fernandez, E. B., Security Patterns and Secure Systems Design.// Fourth LACCEI International Latin American and Caribbean Conference for Engineering and Technology [LACCET´2006]., Breaking Frontiers and Barriers in Engineering: Edication Research and Practice./21-23 June 2006, Mayaguez, Puerto Rico., Conference Paper. September 2007.
30. Hoepman, J., H., Privacy Design Strategies. (Extended Abstract)., In://IFIP AICT 428, pp.446-459, 2014.
31. Hofmann, A., Apfel, B., Barth, U., Günther, Ch., Haas, I., Holzwarth, F., Kramer, A., Kunz, L., Sator, N., Siebert-Cole, E., and Strober, P., Lecture Notes in Computer Science: Authors Instructions for the Preparation of Camera-Ready Contributions to LNCS/LNAI/LNBI Proceedings., Aviable from: http://www.springer.com/journal/13. ISSN: 1420-8938(electronic version).
32. HP, Security Research Cyber Risk Report, 2015, Aviable on: [http://whp-aus2.cold.extweb.hp.com/pub/msc/B9302434-70B4-45CE-AEB7-29F6EAD6E2FE.pdf]
33. Interoperable EU Risk Management Framework, European Union Agency for Cybersecurity (ENISA), 2021, DOI:10.2824/07253
34. ISO/IEC 27001: 2013 Information technology — Security techniques — Information security management systems — Requirements, 2013
35. ISO/IEC 2382-1:1993 Information Technology – Vocabulary – Part 1: Fundamental terms. International Organization for Standardization (ISO). [Online]. Available from [http://www.iso.org/iso/catalogue_detail.htm?csnumber=7229]

36. ISO 23903:2021 Health informatics — Interoperability and integration reference architecture — Model and framework

37. ISO/IEC 27000:2018 Information technology — Security techniques — Information security management systems — Overview and vocabulary

38. ISO/IEC 27005:2018 Information technology — Security techniques — Information security risk management

39. Gressl, L., Steger, Ch.,,Neffe, U., A Security Aware Design Space Exploration Framework. //Conference Paper March 2019/ DOI:10.1109/FDL 2019.8876944, Aviable from: https://www researchgate. net/publication 33202962].

40. Gressl, L., Steger, Ch., Neffe, U., A Security Driven Design Space Exploration for Embedded Systems. //Conference Paper September 2019/ Aviable from: https://www researchgate. net/publication 336728352].

41. Gedeon, I. J., Frey, C., Mohanty, S. P., Sonively, P., Almuhtadi, W., Privacy and Security by Design. In: //IEEE Consumer Electronics Magazine, March 2020, DOI:10.1109/MCE2019.2954762, Available from: https://www researchgate. net/publication 339608556].

42. Gressl, L., Rech, A., Steger, Ch., Sinnhofer, A., Weissnegger, R., Security Based Design Spase Exploration for CPS.// Conference Paper March 2020/ DOI:10.1145/3341105.3374058, Aviable from: https://www researchgate. net/publication 340268505].

43. Guggenmos, F., Hackel, B., Ollig, P., Stahl, B. Security First, Security by Design, or Security Pragmatism-Strategic Roles of IT Security in Digitalization Projects.//Computers $ Security, 118 (2022)102747., Available from: https://www. elsevier.com/locate/cose.

44. Khac Hai N., Lawpoolsri S., Jittamala P., Thi Thu Huong P., Kaewkungwal J (2017) Practices in security and confidentiality of HIV/AIDS patients' information: A national survey among staff at HIV outpatient clinics in Vietnam. PLoS ONE 12(11): e0188160. https://doi.org/10.1371/journal.pone.0188160

45. Koleva, E., E. Andreev, M. Nikolova, 2022.A metagraph model of cyber protection of an information system., Mathematics and Informatics,, 65(2), 201 – 211, https://doi.org/10.53656/math2022-2-7-ame

46. Kolomoitcev, V., A Layered Information Security Sistems Designing.// Conference Paper January 2020, DOI:10.31799/978-5-8088-1452-3-2020-1-190-192, Aviable from: https://www researchgate. net/publication 340701963].

47. Kǿien, G. M. A Philosophy of Security Architecture Design.//Wireless Personal Communications (2020) 113:1615-1639./ https://doi.org/10.1007/s11277-020-07310-5.

48. Krutz, R. L., R. D. Vines (May 2003). The CISM Prep Guide: Mastering the Five Domains of Information Security Management. Wiley. p. 12. ISBN 978-0-471-45598-1.

49. Lazarinis, F., Green, S., Pearson, E. (Eds.), (2011). Handbook of Research on E-Learning Standards and Interoperability: Frameworks and Issues. IGI Global. Avaiable at: [https://doi.org/10.4018/978–1-61692-789-9]

50. Liu, F., Lee R. B., Security Testing of a Secure Cach Design.// Conference Paper June 2013/ HASP´13, June 2013, Tel-Aviv Israel, DOI:10.1145/2487726.2487729, Available from: https://www researchgate. net/publication 262389651].

51. Maguire, M., Westbrook, D.A. Security by Design: Counterterrorism at the Airport. //Anthopology Now, 12:3, 122-135, https://doi.org/10.1080/19428200.2020.1884480.

52. Moganedi, S., Didmini, S., Security by Design: Rethinking Resilience of IoT in Healthcare.// Conference Paper. May 2021/ IST-Africa 2021 Conference Proceedings Miriam Cunningham and Paul Cunningham (Eds)., IST-Africa Institute and IIMC. 2021, ISBN:978-1-905824-67-0., https://www.IST-Africa.org/Conference 2021.

53. Messe, N., Belloir, N., Chiprianov, V., El-Hachem, J., Fleurquin, R., Sadou, S., An Asset-Based Assistance for Secure by

Design.// Conference Paper. December 2020, DOI:10.1109/ APSEC51365.2020.00026, Aviable from: https://www researchgate. net/publication 346240536].

54. Nomikos, K., Papadimitriou, A.,Stergiopoulos, G., Koutras, D., Psarakis, M., Kotzanikolaou, P. On a Security-oriented Design Framework for Medical IoT Devices: The Hardware Security Perspective.//23rd Euromicro Conference on Digital System Design (DSD), 2020, pp.301-308, DOI:10.1109/DSD51259.2020.00056.

55. Pescador, F., Mohanty, S., Guest Editorial Security-by-Design for Electronic Systems. In: //IEEE Transactions on Consumer Electronics/ February 2022, DOI:10.1109/TCE2022.3147005, Aviable from: https://www researchgate. net/publication /359885940].

56. Petrova, V., The Hierarchical Decision Model of cybersecurity risk assessment, 2021, 12th National Conference with International Participation (ELECTRONICA), 2021, pp. 1-4, Electronic ISBN:978-1-6654-4061-5. doi: 10.1109/ELECTRONICA52725.2021.9513722.

57. 12. Petrova, V. Development of secure software. X International Scientific Conference "Technics. Technologies. Education. Safety". 06– 09.06.2022, Borovets, Bulgaria, ISSUE 1 (14), pp. 35-38, ISSN 2535-0315 (Print), ISSN 2535-0323 (Online)

58. Ranum, M., „Thinking about firewalls". *Vtcif.telstra.com.au. 1990-01-20. Archived from* the original *on 2020-01-05*. Retrieved 2012-01-19

59. Rest, J., Boonstra, D., Everts, M., Rijn, M., and Paassen, R., Designing Privacy-by-Design. In://(Eds): APF 2012, LNCS 8319, pp. 55-72, 2014. Conference Paper. January 2014.,DOI:10.1007/978-3-642-54069-14. Available from: https://www. researchgate. net/ publication/300040701].

60. Rocha, A., Martins, A., Paiva Dias, G., Reis, L., Cota, M. Sistemas e Tecnologias de Informacão.//Atas da 10 Conferencia Iberica de Sistemas e Tecnologias de Informacao, Agueda, Portugal, 17-20 Junho 2015/AISTI Universidade de Aveiro, -vol I– Artigos, Tomo 1, ISBN: 978-898-98434-5-5.

61. REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL, Official Journal of the European Union, 4.5.2016, L 119/1

62. Sharma, N., Sahu, R. A., Sarasmat, V., Sharma, B. K., Adaptively Secure Strong Designated Signature.// Conference Paper. December 2016, LNCS 10095,pp 43-60,2016. DOI:10.1007/978-3-319-49890-4_3, Available from: https://www researchgate. net/publication /309819958].

63. Nataliya Shevchenko, Timothy A. Chick, Paige O'Riordan, Thomas Patrick Scanlon, PhD, & Carol Woody, PhD July 2018, THREAT MODELING: A SUMMARY OF AVAILABLE METHODS, SOFTWARE ENGINEERING INSTITUTE | CARNEGIE MELLON UNIVERSITY, Avaiable at: [https://apps.dtic.mil/sti/pdfs/AD1084024. pdf]

64. Siavvas, M., G elenbe, E., Tsoukalas, D., Kalouptsoglou, I., Mathioudaki, M., Nakip, M., Kehagias, D., Tzovaras, D., The IoTAC Software Security-by-Design Platform: Concept, Challenges, and Preliminary Overview.// Conference Paper: March 2022, DOI:10.1109/DRCN53993.2022.9758028, Available from: https:// www researchgate. net/publication /360058675].

65. Shin, Seong-Yoon., A Study on Security A Tribute Design in Security Plan of The Design Phase. In:// The Journal of the Korean Institute of information and Communication Engineering. May 2015, DOI:10.6109/jkiice.2015.19.5.1125, Aviable from: https://www researchgate. net/publication /283202321].

66. Santos, J., Tarrit, K., Mirakhorli, M. A Catalog of Security Architecture Weaknesses. //IEEE International Conference of Software Architecture Workshops, 2017, pp.220-225, DOI:10.1109/ICSAW.2017.25.

67. Salnitri, M., Alizadeh, M., Giovanella, D., Zannone, N., Giorgini, P., From Security-by-Design to the Identification of Security-Critical Deviations in Process Executions. //Chapter: June 2018, DOI:10.1007/978-3-319-92901-9-19, Available from**:** https://www researchgate. net/publication /325603839].

68. Solms, S., Futcher, L. A. Adaption of a Secure Software Development Methodology for Secure Engineering Design.//DOI:10.1109/*ACCESS* 2017. DOI Number June 12, 2020.

69. Standard Computer Dictionary IEEE, A Compilation of IEEE Standard Computer Glossaries. IEEE, New York, NY, 1990, Aviable at: [https://www.standardsuniversity.org/article/standards-glossary/#I].

70. Teimoor, R. A., A Review of Database Security Concepts, Risks, and Problems. In: //UHD Journal of Science and Technology. Jul 2021, vol 5/Issue 2., DOI:10.21928/uhdjst.v5,n2y2021.pp.38-46, Available from: https://www researchgate. net/publication /356421044].

71. Uskov, V. L., Howlett, R., Jain, L. Smart Education and e-Learning 2020.//Smart Innovation, Systems and Technologies. ISBN 978-981-15-5583-1, https://doi.org/10.1007/978-981-15-5584-8.

72. Yazar Z., A Qualitative Risk Analysis and Management Tool – CRAMM, SANS Institute, 2021, Available from [sansorg.egnyte.com/dl/EVhEaxZS8S].

доц. д-р Веселина Спасова

СИГУРНОСТ ПО ДИЗАЙН
В СОФТУЕРНОТО ИНЖЕНЕРСТВО

Българска, първо издание

Рецензенти:
проф. д.н. Борислав Стоянов,
проф. д-р Теодора Бакърджиева
Дизайн на корицата: Николай Иванов

Формат 60/84/16
Печатни коли 7,25
ISBN 978-954-715-834-7

ВСУ „Черноризец Храбър“
Издателски център