iB
INFORMATIEBEVEILIGING
jaargang 16 - 2016
#4

DATA-CENTRIC SECURITY

Agile Security

Security-awareness: zo wordt het een succes

Public key Pinning

Verizon DBIR

Certificeren? Dan moet u voldoen aan de norm. DNV GL toetst u snel en goed. Maar iedereen houdt van opstekers, niet van standjes. Daarom kijken we bij certificering ook naar wat goed gaat en zelfs nog beter kan. Op die gebieden die voor uw bedrijf of organisatie belangrijk zijn. Aandachtspunten waarop u zélf beoordeeld wilt worden. Certificering die net even verder voert. Want verbetering zit in een klein hoekje.

U kunt ons bereiken via 010 2922 700 of www.dnvgl.nl

Stappenplan ISO 27001/NEN 7510

Download kosteloos de whitepaper 'Stappenplan naar informatiebeveiliging'

www.dnvgl.nl/whitepapers

SAFER, SMARTER, GREENER

DNV·GL

SLIMME SOFTWARE

Software wordt tegenwoordig in heel veel producten toegepast. Ik zou eigenlijk geen elektronisch apparaat meer kunnen noemen dat géén software bevat. En dan heb ik het nog niet eens over IoT (Internet of Things)-apparaten…

De software maakt in vele gevallen dat een apparaat beter kan werken. De koffiemachine die ik in mijn keuken heb staan, voert functies en controles uit die mechanisch of elektrisch nauwelijks te regelen zijn. Maar omdat het onder softwarebesturing gaat, kan het allemaal. En we zijn dat soort apparaten 'slimme' apparaten gaan noemen.

Maar hoe goed 'slim' zijn ook is, het kan ook tegen je gebruikt worden. En ik heb het hier niet over malware. Als informatiebeveiligers kennen we malware en weten we hoe slecht dat kan uitpakken. Ik hoor nu al van IT-ers dat ze een uitbuiter van ransomware uiteindelijk maar betaald hebben. Ik wacht nog op de beveiliger die het durft op te biechten.

Slimme apparaten zijn slim genoeg om ons te bedriegen. Het is nu bekend dat Volkswagen de smogtest detecteerde en de vervuiling dan wel zuinigheid aanpaste aan de test. En ook loterijcomputers kunnen geprogrammeerd worden om – na certificatie – een paar dagen van het jaar een voorspelbare uitkomst te geven, in plaats van een random uitkomst. Dit werd echter niet eens ontdekt omdat een inspectie het programma had doorgelicht en bedrog had gevonden maar omdat het opviel dat de casinomanager in kwestie miljoenen won.

En een ander effect: slim falen. Of is dat toch dom falen? USB is zo slim dat een PC iedere keer dat er een apparaat aangesloten wordt weer een andere driver installeert. Wat moet je als 'domme' computereigenaar als je PC ineens meldt dat hij er zelf niet meer uitkomt en er dus een USB-driver-probleem is? Los het maar op. Ik had er nooit van gehoord, kreeg er recentelijk mee te maken en via Google kwam ik erachter dat dit toch wel vaker voorkomt… Ik heb nog steeds geen oplossing.

Dit brengt mij tot de hamvraag, die volgens mij heel belangrijk gaat worden voor professionals in software-assurance: hoe kun je je in deze wereld verzekeren dat software niet alleen doet wat het moet doen, maar ook alles doet wat het moet doen en ook alleen maar doet wat het moet doen? Juist die laatste twee aspecten zijn alleen maar aan te tonen na lange, ingewikkelde analyses en hebben gemaakt dat Volkswagen en de oud-casinomanager zich veilig voelden in hun bedrog.

In dit nummer genoeg over software: agile security, security awareness en de Verizon DBIR.

Lex Borger, hoofdredacteur

In dit nummer

AGILE SECURITY

De herijking van informatiebeveiliging voor agile projecten

Meer en meer systeemontwikkeling vindt plaats op basis van agile methodieken. Het voordeel van agile is dat snel nieuwe functionaliteit toegevoegd kan worden en het risico van complexe wijzigingen sterk wordt gereduceerd door deze op te knippen in kleinere brokken. Klassieke informatiebeveiliging heeft echter moeite met deze agile aanpak. Wat zijn de oorzaken van dit falen en hoe kan een acceptabel beveiligingsniveau in agile ontwikkelomgevingen dan wél worden bereikt?

De mismatch tussen traditioneel securitymanagement en agile ontwikkeling wordt onder meer veroorzaakt door de gebruikelijke manier van werken binnen securitymanagement. Veelal worden hier standaarden en raamwerken toegepast die een top-down-benadering kennen: eerst moet er beleid komen en een organisatie worden ingericht, daarna moeten de procedures worden opgesteld en tenslotte wordt er een verzameling van generieke technische maatregelen bedacht. En pas als dat allemaal 'geregeld' is kunnen projecten gebruik maken van de diensten van securitymanagement.

Deze raamwerken zijn essentieel voor een goed gebalanceerde en beheersbare beveiliging binnen een organisatie. Ze passen echter niet goed bij een agile traject waar een ontwikkelaar op detailniveau wil weten welke maatregelen geïmplementeerd moeten worden. Daarnaast sluit de beschrijving van de maatregelen niet aan bij de taal en cultuur van ontwikkelaars. Veel ontwikkelaars zien beveiliging als een hindernis (de 'paarse krokodil') binnen een agile project die een snelle levering van nieuwe functionaliteit onmogelijk maakt. De snelle ontwikkelmethoden staan haaks op het gebruik van compliance-checklists die niet met hun tijd zijn meegegaan.

Hoe kunnen we dat veranderen? Dit artikel onderzoekt een aantal mogelijkheden om informatiebeveiliging op de juiste wijze te integreren in agile. Door bij de basis van securitymanagement te beginnen en op die basis een nieuwe manier van werken te ontwikkelen kan binnen agile het uiteindelijke doel worden bereikt: een snelle en beheersbare ontwikkeling waarbij ook de beveiligingsrisico's op de juiste en acceptabele wijze zijn geadresseerd.

Het beveiligingsperspectief op agile ontwikkeling

Het doel van de agile aanpak is om zo snel mogelijk nieuwe functionaliteit te implementeren door middel van een continue release van kleine stappen. De agile aanpak probeert niet in één keer de gedetailleerde scope voor het gehele project te definiëren en in detail de producten te beschrijven. Bij agile wordt het project opgeknipt in kleinere, overzichtelijke, stappen (sprints) die beheersbaar en bestuurbaar zijn. De besturing van deze sprints is zodanig dat ontwikkelaars gedwongen zijn keuzes te maken welke functionaliteit wel en niet in een sprint wordt gebouwd. Alle 'feature requests' worden verzameld in een backlog, en voor elke sprint wordt een selectie van deze features gemaakt op basis van gewenste functionaliteit, urgentie, beschikbare middelen, verwachte inspanning en andere criteria. Als een feature-request te groot of complex is, wordt deze opgeknipt in kleinere onderdelen die afzonderlijk kunnen worden geïmplementeerd, al dan niet in een aantal sprints achter elkaar. Testresultaten en bevindingen uit vorige sprints worden als input gebruikt voor nieuwe sprints om zo direct de bevindingen te kunnen oplossen.

De traditionele aanpak van beveiliging is veel meer gericht op het traditionele watervalmodel waarbij beveiliging vaak als eerste en als laatste in de keten aan bod komt. Aan het begin wordt beveiliging ingeschakeld om de eisen en wensen op te stellen, en aan het einde van het traject om te controleren of deze op de juiste wijze zijn geïmplementeerd. Dit model werkt niet meer in agile.

In dit artikel wordt een aantal van de belangrijke aspecten van agile ontwikkeling uitgewerkt die gerelateerd zijn aan de manier waarop een organisatie haar beveiliging heeft ingericht. Deze aspecten zijn:

1. Keuzes: In elke sprint moeten ontwikkelaars continu keuzes maken; wordt een bouwsteen wel of niet gebouwd. Op basis van de gewenste functionaliteit, projectrisico, bijdrage aan het eindresultaat en beschikbaarheid van mensen en middelen worden deze keuzes gemaakt. Informatiebeveiliging zal op dezelfde wijze vergelijkbare keuzes moeten maken. Deze keuzes zijn gebaseerd op de onderkende beveiligingsrisico's. Dit houdt in dat er continu risico-assessments worden uitgevoerd, op het detailniveau van de sprint en daarin gedefinieerde bouwstenen. De risico-assessments dienen ook in uitvoering binnen de planning van de sprint te vallen, het liefst zelfs korter zodat de resultaten al in de sprint kunnen worden meegenomen. Uiteraard zullen

Arthur Donkers is sinds 1995 bezig met informatiebeveiliging. Hij is begonnen in de techniek, van het scannen van netwerken tot het testen van complexe web applicaties en grote infrastructuren. In de loop van de jaren kwam ook het inzicht dat beveiliging meer is dan techniek en ook de proces en organisatorische kant de juiste aandacht verdient. Sindsdien probeert Arthur deze drie domeinen, met risico als bindende factor, met elkaar te verbinden tot oplossingen. Daarbij maakt Arthur gebruik van security architectuur op basis van SABSA, TOGAF en gezond verstand. Arthur is te bereiken via arthur@1secure.nl.

Pascal de Koning is werkzaam bij Ideas to Interconnect en bereikbaar via p.de.koning@i-to-i.nl.

de beveiligingsmaatregelen ook binnen de sprint moeten passen. Dit houdt in dat ze op het juiste detailniveau gedefinieerd moeten zijn en zo specifiek mogelijk, zodat zij binnen de schaal en planning van de sprint passen.

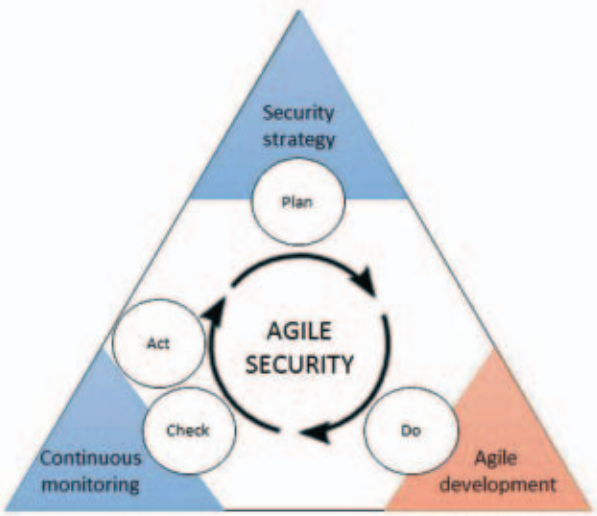
2. Bottom-up: Binnen agile worden producten vaak via een bottom-up-aanpak ontwikkeld waarbij kleine bouwstenen eerst worden ontwikkeld die vervolgens door andere bouwstenen gebruikt kunnen worden. Het breidt uit van onderaf. In de klassieke watervalmethode wordt juist een top-down-besturing gevolgd. Ieder componentje krijgt een plaats in een van tevoren uitgedacht geheel. De meeste beveiligingsraamwerken zijn op dit model gebaseerd. Om beveiliging aan te laten sluiten bij agile moet ook binnen beveiliging een bottom-up-aanpak worden ingevoerd. Beveiligingsbouwstenen en maatregelen moeten aansluiten bij de agile benadering, het juiste detailniveau hebben en passende beveiligingsdiensten bieden..

3. Snelheid: agile ontwikkeling levert kleine, incrementele stappen in functionaliteit in korte sprintcycli. Deze korte tijd betekent dat securitymanagement niet uitgebreid de tijd heeft om allerlei standaarden en beleid te formuleren, laat staan deze uitgebreid te controleren. Verder betekent deze incrementele oplevering dat er niet een vast gedefinieerd product wordt opgeleverd in een sprint: functionaliteit kan nog ontbreken. Voor beveiliging betekent dit dat er geen uitgebreide (klassieke) beveiligingstesten gedaan kunnen worden uitgevoerd. Hiervoor is de tijd niet beschikbaar en het te testen systeem is te veel in beweging. De beveiligingscontroles zullen ook op incrementele wijze moeten worden uitgevoerd en de sprintcycli moeten volgen. Hierbij wordt het bijhouden van een risicoregister, met de juiste scope en details, van essentieel belang. Dit risicoregister is de backlog voor beveiliging.

4. Terugkoppeling: dankzij de korte cyclustijd van een sprint, is het mogelijk bevindingen van vorige sprints mee te nemen in volgende sprints. Dit model leidt uiteindelijk tot een proces van continue verbetering. In het watervalmodel was het mogelijk dat securitymanagement een release kon blokkeren vanwege bijvoorbeeld kritieke risico's. Binnen het waterval model is er immers een duidelijk moment van vrijgave te definiëren. Binnen agile is er niet zo'n duidelijk moment, de incrementele releases volgen elkaar op in een continue stroom. Beveiliging heeft hierdoor wel de mogelijkheid eerder kritieke risico's te identificeren en deze te laten oplossen in een sprint waardoor de kans op blokkade aan het einde van een project aanzienlijk wordt verminderd. Door in een sprint op de juiste risico's te toetsen is beveiliging, mits zij aansluit bij agile, in staat de juiste focus te leggen en de juiste risico's te identificeren.

Waarom wringt ISO27001 binnen agile? Het securitymanagementproces, met de PDCA-cyclus en ISMS

In veel organisaties wordt ISO27001 (of een afgeleide daarvan) toegepast om richting en sturing te geven aan de informatiebeveiliging en de bijbehorende beheerprocessen. Het ISO27001-raamwerk is gebaseerd op een ISMS en bijbehorende Plan-Do-Check-Act-cyclus (overigens heeft de nieuwste versie van ISO27001 (de 2013 versie) de PDCA-cyclus min of meer losgelaten en is de focus meer op de context van de organisatie komen te liggen). De PDCA-cyclus begint met een risico-assessment en gebruikt daarvoor een top-down-benadering om de juiste doelstellingen te definiëren. Op basis van de doelstellingen worden maatregelen gekozen die de geïdentificeerde risico's moeten adresseren. Deze maatregelen bestaan vaak uit een combinatie van maatregelen op organisatorisch, procesmatig en technisch niveau. De effectiviteit van deze maatregelen wordt vervolgens gemeten en op basis daarvan wordt het beveiligingsplan bijgesteld. Dit is heel kort samengevat de PDCA-cyclus die op basis van een top-down-benadering beveiliging inricht.



Figuur 1 - De planning- en control-cyclus binnen securitymanagement

En alhoewel ISO27001 een veel gebruikt raamwerk is dat een organisatie in staat stelt om haar beveiligingsbehoefte te beheren, past ISO27001 niet eenvoudig op een agile ontwikkelproces. De top-down-benadering, samen met het feit dat veel maatregelen te generiek zijn, zorgt voor een gat tussen het raamwerk en de daadwerkelijke behoefte binnen agile.

Aannames en waar zij falen voor agile

De klassieke raamwerken doen een aantal aannames over de manier waarop securitymanagement integreert met systeemontwikkeling:

1. Het projectteam is in staat de generieke beveiligingseisen te vertalen naar specifieke beveiligingsmaatregelen;
2. Het projectteam heeft de tijd, kennis en middelen om deze maatregelen op de juiste wijze te implementeren;
3. Er is voldoende tijd en budget om in het project een technische beveiligingstest uit te voeren en de bevindingen te verwerken;
4. Er is voldoende tijd om alle beveiligingsrisico's te identificeren en op te lossen.

Deze aannames gaan mank in een agile ontwikkeltraject.

1. Het ontwikkelteam heeft andere prioriteiten en beperkte middelen om alle eisen te vertalen naar maatregelen en deze vervolgens te implementeren. Binnen agile is timemanagement essentieel en moeten continu keuzes worden gemaakt. Helaas zijn deze keuzes vaak nadelig voor beveiliging.

2. Het ontwikkelteam heeft niet de kennis en ervaring om beveiligingsmaatregelen te ontwikkelen en definiëren. Zij zijn zich niet altijd bewust van de kwetsbaarheden in bouwstenen en hebben vaak geen kennis van methodieken om veilig software te ontwikkelen.

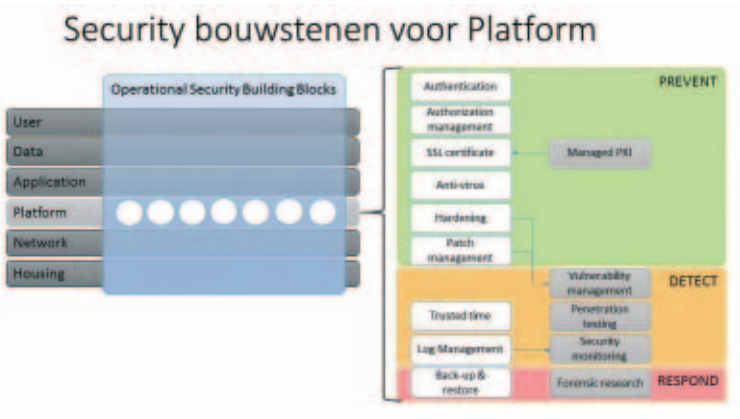
3. Binnen de ontwikkelcyclus is vaak geen duidelijke testfase aanwezig waarin de functionele en niet-functionele eisen worden getest. Testen vindt veel meer plaats op unit-niveau en richt zich primair op functionele testen. Hierdoor blijven testen op beveiliging vaak achterwege, ook omdat er binnen een sprint geen tijd is voor securitymanagement om uitgebreide testen te doen. Bovendien is de focus van een sprint vaak pas bekend bij aanvang van de sprint zelf en heeft securitymanagement niet de tijd en middelen om hierop in te spelen.

4. Tenslotte zijn er culturele issues. De meeste ontwikkelaars krijgen geen warm gevoel bij informatiebeveiliging. Het beeld is dat beveiliging een hoop overhead creëert omdat verplicht allerlei maatregelen moeten worden geïmplementeerd. Deze maatregelen kosten een hoop tijd, vertragen de ontwikkeling en geven niet direct een aantoonbaar resultaat.

Agile securitymanagement

Het doel van informatiebeveiliging is om de risico's tot een acceptabel niveau te beperken, zodat de doelstellingen van de bedrijfsprocessen kunnen worden gehaald.

De business- of systeemeigenaar wil in control zijn en heeft daarbij securitymanagement nodig. Om dit te bereiken wordt de Plan-Do-Check-Act-cyclus vaak toegepast. Wanneer deze (strategische) PDCA-cyclus wordt toegepast in een agile omgeving, past het agile ontwikkelproces het best in de Do-fase. Dat is waar de maatregelen worden geïmplementeerd. Het securitymanagementproces zorgt voor inbedding van beveiliging in agile ontwikkeling; door het stellen van eisen (Plan), door het selecteren van de juiste maatregelen (Do), door het monitoren van beveiliging (Check), en het aanpassen naar aanleiding van nieuwe eisen of wijzigingen (Act). Dit is weergegeven in Figuur 2. Deze positionering van agile in de PDCA-cyclus is het gevolg van de bovengenoemde vier aannames. Iedere fase uit de PDCA-cyclus wordt verder besproken in de volgende paragrafen.



Figuur 2 - Agile securitymodel, waarin agile development is geplot op de Plan-Do-Check-Act-cyclus

Plan: Securitystrategie en governance

Security-governance in een agile omgeving is gelijk aan traditionele security-governance. Deze component zorgt ervoor dat de PDCA-cyclus wordt uitgevoerd. Ten aanzien van requirements-management houdt het bij wat de wettelijke, branchegebonden en organisatiespecifieke eisen rond security zijn. Tevens houdt het bij welke risico's zijn geïdentificeerd gedurende het agile ontwikkelproces en of deze in lijn zijn met de security-doelstellingen. In een agile omgeving zorgt security-governance door de bestaande audit- en GRC-processen in lijn te brengen met agile ontwikkeling, zowel in scope als detailniveau.

Do: Secure agile ontwikkelproces

Het agile ontwikkelproces is waar het rubber het asfalt raakt, met een actieve bijdrage van securitymanagement in de sprints.

Aanname 1: niet in staat om requirements naar security-maatregelen te vertalen

De eerste aanname was dat het agile projectteam niet in staat is om de security-doelstellingen door te vertalen naar security-maatregelen, door gebrek aan tijd, geld of de juiste mensen.

Dit vereist actieve betrokkenheid van één of meer medewerkers uit het security-management-domein in het project. Ze zijn deel van het team en ondersteunen het project om de security-requirements en functionele requirements tegen elkaar af te wegen indien nodig. De securitymedewerkers dienen kennis te hebben van de verschillende domeinen en technieken. Ze selecteren de juiste beveiligingsmaatregelen die relevant zijn voor de functionaliteit die op dat moment door het ontwikkelteam wordt gerealiseerd. Op deze manier wordt het gat overbrugd tussen het generieke security-raamwerk op organisatorisch niveau en de specifieke security-maatregelen die op projectniveau nodig zijn.

De vertaling van security-doelstellingen naar beveiligings-maatregelen kan worden gestandaardiseerd door:

- De definitie van een security-baseline, welke een lijst bevat van alle beveiligingsmaatregelen die dienen te worden geïmplementeerd;
- De definitie van een security-classificatieschema. Wanneer een sprint begint wordt de bouwsteen in kwestie geclassificeerd voor bijvoorbeeld vertrouwelijkheid, privacy en toegankelijkheid. De classificatie leidt tot een set van beveiligingsmaatregelen die het juiste niveau van bescherming bieden. Het classificatieschema zorgt ervoor dat de baseline wordt afgestemd op verschillende beveiligingsniveaus.

Aanname 2: gebrek aan expertise om beveiligingsmaatregelen te implementeren

De tweede aanname was dat het agile ontwikkelteam onvoldoende expertise heeft om de beveiligingsmaatregelen correct te implementeren. De rol van het security-team is daarom om het implementeren van de maatregelen zo eenduidig en eenvoudig mogelijk te maken door ze als kant-en-klare security-bouwstenen aan te leveren. Deze kunnen snel worden geïntegreerd in het ontwikkelende systeem.

Security-bouwstenen kunnen de vorm hebben van:

- Operationele securityservices die as-is kunnen worden geïntegreerd in het systeem. Een voorbeeld is een authenticatieservice. De generieke services in de servicecatalogus beschrijven een authenticatieservice met verschillende beveiligingsniveaus, variërend van wachtwoorden tot twee-factor-authenticatie met certificaten. Alle systemen implementeren minimaal het laagste niveau (met de wachtwoorden). Dit kan fysiek zijn geïmplementeerd door een Active Directory. Voor sterkere authenticatie kan bijvoorbeeld twee-factor-authenticatie beschikbaar zijn middels een federated service op basis van SAML. De verwijzing naar de juiste fysieke service wordt gedaan door de servicecatalogus in combinatie met de security-baseline.
- Security-patronen: standaardmanieren om standaardissues mee op te lossen. Een voorbeeld is de manier waarop patch-management wordt gedaan. Hoewel de frequentie van patch-releases en hoe deze toe te passen verschilt per technologieleverancier, kan de generieke procedure voor het toepassen van patches dezelfde zijn voor alle systemen in de organisatie.

De verzameling van kant-en-klare security-bouwstenen kan worden gevonden in de securityservices-catalogus. De inhoud van deze catalogus zal voor iedere organisatie verschillend zijn. In het beste geval is het gebaseerd op een praktische catalogus van operationele securityservices, welke zou kunnen worden gegeven door branchestandaarden en platformspecifieke security-standaarden. De services in de catalogus zijn geordend naar de laag in de system-stack waarin ze opereren, zoals weergegeven in onderstaande diagram.

Figuur 3 - Voorbeeld van een baseline met verplichte securityservices op de platform-laag

Zoals is te zien in de diagram hebben bepaalde services een preventief karakter, andere juist een detectie of correctief karakter. Welke services deel uitmaken van de baseline hangt af van de context, beschikbaar budget en risk-appetite van de organisatie. Het systeemontwikkelproject hoeft alleen de witte services uit het diagram te implementeren: de grijze services worden geleverd door securitymanagement.

Check: Continuous security monitoring

Security Monitoring is een breed begrip. Doorgaans wordt hieronder verstaan het monitoren op operationele security-events. Met betrekking tot agile systeemontwikkeling gaat het om het monitoren of er veilige systemen worden opgeleverd, het monitoren op events is even buiten de scope gehouden. Waar het om gaat is dat als securitymanagement de snelle cycli van agile development niet bij kan houden, het hele idee van synchroniseren van agenda's moet worden losgelaten. Het onderzoeken van kwetsbaarheden zou volledig onafhankelijk van het ontwikkelproces moeten worden uitgevoerd.

Aanname 3: er is geen tijd voor security tests in de sprint

De agile aanname is dat er geen tijd voor het plannen en uitvoeren van een kwetsbaarhedenonderzoek voordat de functionaliteit wordt opgeleverd; het nieuwe (deel)systeem wordt hoe dan ook opgeleverd. Als dit het geval is, waarom zouden we er dan niet van uitgaan dat securitymanagement überhaupt niet wordt geïnformeerd van enige nieuwe systeemrelease. Het moet zelf maar uitzoeken welke systemen erbij zijn gekomen, en het moet de veiligheid van deze systemen controleren ongeacht de methode waarmee ze tot stand zijn

gekomen. Dat is continuous monitoring in een agile omgeving: bijhouden welke systemen er zijn en de veiligheid checken onafhankelijk van het ontwikkelproces. Het vaststellen van kwetsbaarheden in systemen is dus niet langer planning-gedreven (de releasedatum van het systeem is immers onbekend), maar observering-gedreven (er is een nieuw systeem gedetecteerd, dus direct gaan testen).

In de praktijk begint continuous monitoring met discovery-scans op het netwerk. Wanneer een nieuw systeem wordt aangetroffen wordt automatisch een check op kwetsbaarheden gestart. Dit kan op verschillende manieren:

- Een geautomatiseerd kwetsbaarhedenonderzoek (door de bouwsteen 'Vulnerability management');
- Een penetratietest door een security-expert (bouwsteen 'Penetratietesten').
- Een consistentie-check op implementatie van de security-baseline (door bouwsteen 'Security-monitoring'): zijn alle verplichte security-bouwstenen geïmplementeerd?

Deze laatste behoeft enige toelichting. Het vereist dat er een set security-bouwstenen wordt gebruikt in de organisatie, welke in samenhang opereren en die alle loggen naar een centraal logsysteem. Stel je voor dat de baseline in Figuur 3 voorschrijft dat een systeem zowel een NTP (bouwsteen 'Trusted time') als Active Directory (specifieke 'Authenticatie' bouwsteen) implementeert. Zodra één van de security-bouwstenen over dit nieuwe systeem een logregel wegschrijft, wordt het ontdekt. Nu komt de consistentie-check: als er een logregel komt uit Active Directory, dan moet er ook een NTP-logregel over dit systeem komen. Zo niet, dan is de conclusie dat de baseline niet volledig is geïmplementeerd door het systeem, en dat kan dan weer leiden tot sancties. Voor deze geavanceerde manier van monitoring is het gebruik van een securityservice-catalogus vereist, evenals een goed werkende security-monitoring-functie.

Act: De cirkel sluiten

Het doel van de Act-fase is om de bestaande situatie bij te stellen, zodat risico's worden gereduceerd tot een acceptabel niveau.

Aanname 4: geen tijd om kwetsbaarheden te herstellen

De vierde aanname was dat er geen tijd is om de security-risico's te mitigeren die tijdens de tests zijn aangetroffen. Er van uitgaande dat de sprints geen ruimte laten voor herstelwerkzaamheden voordat de functionaliteit wordt opgeleverd, dienen andere routes te worden bewandeld om bij te sturen en de kwetsbaarheden weg te nemen. Bijsturing kan worden bereikt op twee manieren:

a) Door het informeren van het security-governance-team welke zich met de Plan-fase bezighoudt. Dit resulteert in een nieuwe security-strategie en bijbehorend security-plan. Dit is de langzame route die leidt tot resultaat op de lange termijn voor een brede groep systemen.

b) Door het informeren van het agile development-team welke zich met de Do-fase bezighoudt. Dit leidt gericht tot veiligere systemen en vermindering van het operationele risico. Dit is de snelle route, maar de scope van de verbeteringen is beperkt tot het systeem in kwestie. Het vereist dat er wordt ingespeeld op de werkwijze van het agile ontwikkelproces. De testresultaten worden geformuleerd als wijzigingsverzoeken en toegevoegd aan de backlog. Ze zullen dan worden overwogen voor de volgende sprint.

Conclusie

De aannames en randvoorwaarden die golden in een traditioneel systeemontwikkelproces zijn niet langer geldig in een agile ontwikkelomgeving. In een agile ontwikkelomgeving dient de security-benadering daarom ook agile te zijn. De belangrijkste veranderde aannames en hun consequenties voor securitymanagement zijn:

Verwacht niet dat het ontwikkelteam de beveiligingsdoelstellingen vertaalt naar de juiste security-maatregelen zonder de juist hulp, ondersteuning en middelen vanuit securitymanagement.

- Stel een snelle en eenvoudige methode beschikbaar voor het classificeren van het systeem. De classificatie fungeert als een security-baseline.

Verwacht niet dat er middelen zijn voor het ontwerpen en implementeren van de beveiligingsmaatregelen.

- Stel operationele security-bouwstenen beschikbaar die klaar voor gebruik zijn. Ze versnellen het ontwikkelproces en zorgen er tegelijkertijd voor dat de maatregelen correct worden geïmplementeerd.

Verwacht niet dat veiligheidsonderzoeken kunnen worden gepland binnen de agile ontwikkelcycli

- Implementeer observatiegedreven security-monitoring, onafhankelijk van het ontwikkelproces: monitor de omgeving continu en controleer de veiligheid van systemen op het moment dat ze zich vertonen.
- Maak gebruik van een samenhangende set van security-bouwstenen. In combinatie met security-monitoring kan hieruit worden geconcludeerd of de security-baseline correct is geïmplementeerd.

Verwacht niet dat er tijd is om kwetsbaarheden weg te nemen voordat het systeem wordt opgeleverd.

- Participeer in het beheer van de backlog, zodat security-aanpassingen worden geïmplementeerd in de volgende sprint.

Dit artikel is onderdeel van een reeks waarin door beide auteurs wordt verkend hoe informatiebeveiliging kan worden geïntegreerd in agile ontwikkeltrajecten. Deze reeks is beschikbaar op http://www.agilesecurity.nl.

SECURITY-AWARENESS: ZO WORDT HET EEN SUCCES

"Waarom moet ík alert zijn op phishing-mails? Dat is toch de taak van de IT-afdeling?" Ze beseffen het zelf niet altijd, maar medewerkers hebben een cruciale rol in de informatie-veiligheid van een organisatie [1, p.55][2]. Een organisatie kan uitmuntende technische en organisatorische maatregelen nemen, maar als medewerkers laconiek met de hen toevertrouwde informatie omgaan, heb je daar alsnog weinig aan. Door te klikken op malafide bijlagen van e-mails, vertrouwelijke werkdocumenten naar privé-mail te sturen en voor alle applicaties hetzelfde wachtwoord te gebruiken -om er maar een paar te noemen- veroorzaken medewerkers grote security-risico's voor hun werkgever. Om die reden hebben de meeste organisaties van een zekere omvang een security-awareness-programma opgetuigd.

Helaas leveren deze niet vaak het gewenste resultaat op [3, p.9]. Na gebombardeerd te zijn met goedbedoelde posters, mailings en e-learnings, halen de meeste medewerkers hun schouders op en gaan ze verder met waar ze gebleven waren. Waar gaat het mis? En vooral: hoe zorg je ervoor dat medewerkers in jouw organisatie wél informatieveilig werken? Lees hier de tien strategie-tips.

1. Begin met het topmanagement

Als directieleden niet bereid zijn om het juiste voorbeeld te geven, dan gaat de rest van de organisatie zijn gedrag ook niet aanpassen. Helaas loopt het hier vaak al spaak. Veel directieleden vinden dat (security)regels wel voor anderen gelden, maar niet voor zichzelf. Het topmanagement is zelfs mogelijk zélf het grootste security-risico, als je het rapport van Stroz Friedberg mag geloven [4]. Zij sturen automatisch werkberichten door naar Gmail. Of ze weigeren beveiligd te printen. Of ze reageren uiterst ontstemd als de receptiemedewerker naar hun toegangspas vraagt. Toch zijn directieleden net gewone mensen en zijn er doelmatige manieren om hen te bereiken. In eerste instantie door de security-boodschap aan te laten sluiten met de organisatiedoelen. Dit toont voor directieleden de relevantie van het thema en geeft direct ook een logische focus aan het programma. En de overige tips in dit artikel zijn ook, in meer of mindere mate, op directieleden van toepassing. Kijk vooral naar de punten 3, 5, 6, 7 en 8.

2. Wat wil je bereiken?

Denk goed na over wat je eigenlijk wilt van de medewerkers. Veel awareness-programma's proberen een set regels over te brengen, met als doel dat de medewerkers zich daaraan houden. De vraag is of dat werkt.

Tegenwoordig hebben de meeste organisaties complexe taken, zijn ze informatie-intensief en bewegen ze zich in een voortdurend veranderende omgeving. Medewerkers zijn gewend om zelfstandig te navigeren in complexe situaties. Een heel specifieke omschrijving van het gewenste gedrag past niet bij de manier waarop medewerkers werken. En, ook relevant, regels zijn statisch, terwijl nieuwe situaties kunnen vragen om ander gedrag. Kwaadwillenden bedenken voortdurend nieuwe methoden (digitaal of 'fysiek') om organisaties waardevolle zaken te ontfutselen. Alertheid en kritisch denkvermogen met betrekking tot informatieveiligheid is in de regel waardevoller dan braaf opvolgen van regels.

3. Ga uit van de medewerker

Idealiter komt het woord 'security' helemaal niet voor in een security-awareness-programma. De focus ligt op de doelgroep: de medewerkers van de organisatie. Hun dagelijkse werkwereld, en dilemma's die daarbij komen kijken, zouden centraal moeten staan. Nu gebeurt dat zelden. Uitgangspunt is meestal het informatiebeveiligingsbeleid plus de bijbehorende richtlijnen. Daar worden alle interventies omheen georganiseerd. Het nadeel hiervan is dat medewerkers de relatie tussen de security-awareness-interventies en hun eigen werk amper zien [5, p.23]. "Ik heb hier helemaal geen tijd voor, ik moet mijn werk doen" is een veelgehoorde opmerking. Het is aan de trekker van het programma om die brug te slaan. Verdiep je in je medewerkers: wat doen ze dagelijks? Wat is hun informatiepositie? Waar werken ze? Welke middelen gebruiken ze? Tegen welke (informatie-)dilemma's en risico's lopen ze aan? Waarschijnlijk kun je zo meerdere doelgroepen onderscheiden. Probeer je boodschap te personaliseren per doelgroep. Een boodschap die persoonlijk, gericht en relevant is, krijgt betekenis voor de toehoorder. De kans is veel groter dat deze er dan iets mee gaat doen [6, p.14, 17].

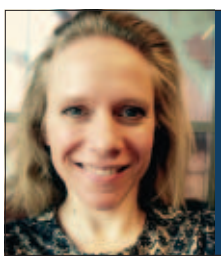
4. Kijk eens naar barrières

Barrières zijn externe factoren die gedrag belemmeren. Zoals security-procedures die dermate omslachtig zijn dat ze (onveilige) short-cuts in de hand werken. Of het ontbreken van veilige alternatieven voor handige maar onveilige middelen.

Alleen het uitdragen van een boodschap is niet voldoende om je doel te bereiken. Medewerkers moeten ook gefaciliteerd worden. Informatieveilig werken zou zo laagdrempelig en vanzelfsprekend als mogelijk moeten zijn. En, in relatie hiermee: maak informatieveiligheid in je organisatie niet onnodig afhankelijk van het gedrag van medewerkers.

Wat voorbeelden:

- Handige maar onveilige online toepassingen zoals Gmail, WhatsApp, Dropbox en WeTransfer zijn mateloos populair bij medewerkers [7]. Verwacht niet dat zij deze gedag zeggen, als de organisatie geen veilige, en net zo handige, alternatieven aanbiedt.
- Verwacht niet van medewerkers dat zij de deur voor de neus van anderen dichtgooien. Dat voelt erg onbehaaglijk voor mensen, die van nature geneigd zijn tot harmonieus sociaal contact [8,



Marijke Stokkel is socioloog en sr. consultant security & risk management bij Ordina. Zij is gefascineerd door de relatie tussen security en menselijk gedrag. Marijke is te bereiken via marijke.stokkel@ordina.nl en Twitter: @marijkestokkel.

Veel directieleden vinden dat regels wel voor anderen gelden, maar niet voor zichzelf

p.73 e.v.]. Toch is deze richtlijn onderdeel van mening security-awareness-programma. Als het voor de organisatie écht belangrijk is dat onbevoegden geen toegang krijgen, kunnen beter tourniquets geplaatst worden. Als iemand tegen ze aan gaat staan in een poortje van een draaideur, zijn mensen wél geneigd om diegene aan te spreken, want dat voelt uitermate vreemd.

5. Besef: mensen zijn geen rationele wezens

Veel programma's werken (impliciet) vanuit de gedachte dat menselijk gedrag gebaseerd is op rationele afwegingen, en dat mensen op basis van goede argumenten hun gedrag aanpassen. Dat blijkt achterhaald [8 p.73 e.v.]. Mensen zijn helemaal geen homo economicus. Allerlei andere factoren spelen ook een rol, zoals -naast de eerdergenoemde barrières- persoonlijke drijfveren, heuristieken en sociale normen [8]. Deze factoren hangen op een complexe manier met elkaar samen en beïnvloeden het (informatieveilige) gedrag van mensen [6].

- Drijfveren zeggen iets over wat we écht belangrijk vinden in het leven [9][10]. Ze hebben te maken met onze motivatie en waarden. Voorbeelden zijn autonomie, creativiteit, harmonie, prestige. Rekening houden met drijfveren die van invloed kunnen zijn op informatieveilig gedrag, kan een programma veel effectiever maken. Harmonieus ingestelde mensen zullen anderen niet snel aanspreken op onveilig gedrag. Mensen die belang hechten aan prestige praten mogelijk sneller hun mond voorbij over een belangrijk vertrouwelijk project.
- Sociale normen gaan over het gedrag van mensen in onze omgeving [8, p.62-65][11]. Wat wij doen, wordt voor een groot deel bepaald door wat anderen doen, vaak meer dan we ons zelf realiseren. 'Als anderen bepaald gedrag vertonen, zal het wel goed zijn', is de (impliciete) gedachte. Sociale normen hebben een grote impact op het gedrag van mensen. Wellicht is het je weleens opgevallen: vaak als iemand ergens van wordt beschuldigd, is het verweer: iedereen doet het!
  Bijvoorbeeld oud-wethouder Jos van Rey, die meldde dat lekken uit de vertrouwenscommissie ('klankborden') bij de benoeming van een burgemeester, schering en inslag is [12].
  Een awareness-programma kan werken met 'goede voorbeelden' van collega's, zodat medewerkers zich daaraan kunnen spiegelen. Het inzetten van ambassadeurs (zie punt 7) kan hierin een rol spelen. Het kan ook gesprekken tussen collega's over informatieveiligheidsdilemma's faciliteren, bijvoorbeeld 'fysiek' tijdens een afdelingsoverleg of online via een forum of app dat wordt gehost op het bedrijfsnetwerk.
- Heuristieken zijn min of meer onbewuste mentale vuistregels. Het zijn denk-strategieën die het verwerken van informatie versimpelen. We hebben ze nodig in het dagelijkse leven, anders zouden we stil staan bij elke handeling. Dat zou te veel tijd en energie kosten. Heuristieken kunnen zowel positieve als negatieve invloed op ons gedrag hebben. Het complexe aan heuristieken is, dat ze veelal onbewust worden toegepast en dat ze moeilijk te veranderen zijn [8, p.21-23].

Voorbeelden zijn:

- 'Als ik gemakkelijk een voorbeeld kan bedenken, zal het wel kloppen' (beschikbaarheidsheuristiek) [13];
- Hoe levendiger de gebeurtenis, hoe beter het is opgeslagen in het geheugen. Levendige gebeurtenissen zijn aansprekend, concreet en nabij (vividness-effect) [14];
- Verschillende conclusies trekken uit dezelfde informatie, afhankelijk hoe de informatie gepresenteerd wordt (framing-effect);
- Te optimistische verwachtingen: overschatten van positieve resultaten (optimism-bias) [13].

Je kunt hiervan gebruik maken door, bijvoorbeeld, veel voorbeelden te geven van informatieveilig gedrag en deze voorbeelden zo levendig mogelijk te beschrijven. En door extra aandacht te besteden aan de informatieveiligheidsrisico's, omdat mensen uit zichzelf geneigd zijn deze te onderschatten.

6. Giet je boodschap in een verhaal

Storytelling is hot! En niet voor niets. Onze hersenen zijn ingesteld op verhalen, ze beklijven veel beter dan losse feiten en cijfers. Zelfs als we informatie krijgen die niet in verhaalvorm verteld wordt, proberen onze hersenen er tóch een verhaal van te maken. Met het vertellen van verhalen proberen we grip te krijgen op een complexe wereld. Goede verhalen spreken niet alleen de ratio, maar ook het gevoel aan. Ze inspireren ons. Dat maakt de kans groter dat we er daadwerkelijk iets mee gaan doen [15].

Een verhaal dient de volgende elementen te bevatten:

- Het draait om een hoofdpersoon
- Het speelt zich in een bepaalde tijd af
- Het heeft een begin- en midden en een eind
- Het heeft een ontwikkeling als gevolg van een worsteling, dilemma of conflict
- Het is authentiek en persoonlijk [16, 17]

Storytelling is uitermate geschikt om dilemma's van medewerkers over te brengen. Bijvoorbeeld: een klant of een bekende in de privésfeer verzoekt een medewerker om bepaalde gevoelige (bedrijfs)informatie te delen. Eigenlijk mag de medewerker niet

voldoen aan het verzoek, maar op de een of andere manier voelt hij zich toch verplicht naar die persoon. Of ziet hij het risico niet van het delen.

Het is belangrijk dat het dilemma van de hoofdpersoon herkenbaar is voor andere medewerkers, zie het kader 'Herkenbaar dilemma'. Het mooiste zou zijn, als je verhalen kunt vertellen die écht gebeurd zijn (geanonimiseerd), en die in werkelijkheid ook nog aanzienlijke consequenties hebben gehad. Dat spreekt de doelgroep het meest aan.

Herkenbaar Dilemma
Vertrouwelijke informatie verkopen aan criminelen is geen goed voorbeeld: het overgrote deel van de medewerkers voelt zich niet aangesproken. Mogelijk zijn zij zelfs beledigd dat zij op deze manier worden bejegend. En die mogelijke uitzondering die wél overweegt om een dergelijke misdaad te begaan, houd je niet tegen met storytelling. Daarvoor moet je andere maatregelen in huis hebben. Je kunt dus beter voorbeelden uit het grijze gebied nemen: die nét niet toegestaan zijn, maar die wel ernstige consequenties kunnen hebben.

7. Zet ambassadeurs in

De boodschapper heeft grote invloed op de ontvangst van de boodschap, mogelijk meer dan de informatie zelf [6, p.16]. Denk daarom goed na wie de boodschap overbrengt. Maak gebruik van ambassadeurs in de organisatie. Dit hoeven niet per se leidinggevenden te zijn; de positie van de boodschapper heeft, zo blijkt uit onderzoek, weinig effect op de impact van de boodschap. Zelfs expertise van de boodschapper is niet zo relevant. Wat wel belangrijk is: betrouwbaarheid, aantrekkelijkheid en gelijksoortigheid met de doelgroep [18, p.51].

8. Intervenieer 'just in time'

Breng je boodschap, als het even kan, op het moment dat de medewerker met het thema bezig is. Dan kan hij of zij direct de nieuwe kennis toepassen en is de kans groter dat het beklijft. In formele leersettings (klasjes, maar ook e-learnings) vergeten medewerkers een groot deel van de stof direct na afloop, omdat ze deze niet direct kunnen toepassen [19].
Bedenk welke momenten medewerkers risico-afwegingen (zouden moeten) maken rond het werken met informatie. Breng ze op die momenten in aanraking met de boodschap van het programma. Just-in-time-interventies kunnen bij uitstek geautomatiseerd plaatsvinden. Geschikte kanalen zijn intranet, diverse sociale media, apps (van de organisatie), sms, e-mail.

Suggesties:
- Stuur eens rond lunchtijd een bericht: hoe laat jij jouw werkplek achter? Eventueel vergezeld van een aansprekende infographic of foto die relevantie van het thema laat zien.
- Toon automatisch een bericht als een medewerker een e-mail gaat sturen naar iemand buiten de organisatie. Dit wordt al door meerdere organisaties toegepast.
- Als een medewerker zijn wachtwoord wijzigt, toon dan automatisch tips rond goed wachtwoordgebruik.
- Zorg ervoor dat online (via het bedrijfsnetwerk) informatie over informatieveiligheid beschikbaar is, overzichtelijk geordend en toegankelijk geschreven. Laat regelmatig weten dat deze informatie beschikbaar is. Als medewerkers gaan 'googlen' op het moment dat ze ergens mee zitten, komen ze vanzelf op de juiste plek terecht.

9. Meet het gedrag

Zorg dat je de resultaten van je programma meet, anders weet je niet of je op de juiste weg zit. Meet bij voorkeur het gedrag van medewerkers. Het sturen van gefingeerde phishing-e-mails is een uitstekende methode: je meet daadwerkelijk gedrag. Minder goede methodes zijn het meten van kennis of houding. Kennis kan namelijk snel weer vervliegen (zie punt 8) en leidt niet automatisch tot beter gedrag [20, p.10].

Goede meetmethoden zijn:
- Gefingeerde social-engineering-aanval:
- Het aantal medewerkers dat meegaat in een (door de organisatie gefingeerde) phishingaanval, door te klikken op een link/bijlage of door waardevolle informatie te verstrekken.
- Het aantal medewerkers dat een phishingaanval weet te stoppen en te rapporteren.
- Het aantal medewerkers dat meegaat in een aanval via mystery calling (telefonisch verzoek om inloggegevens of andere waardevolle informatie).
- Het aantal medewerkers dat een aanval via mystery calling weet te stoppen en te rapporteren.
- Het aantal gestolen of verloren laptops of usb-sticks. Te achterhalen via de incidentregistratie.
- Het aantal met malware geïnfecteerde laptops/werkstations. Ook te achterhalen via de incidentregistratie.
- Het aantal medewerkers dat sterke wachtwoorden gebruikt. Te meten via een brute-forcing-aanval of een poging tot het kraken van de wachtwoord-hashes [21].

Het mooie is, dat je de uitkomsten van bijvoorbeeld phishing en mystery calling ook kunt gebruiken als interventie. Koppel de resultaten binnen zeer korte tijd terug aan de betreffende medewerkers, met adviezen hoe ze een aanval kunnen herkennen en stoppen. En communiceer de resultaten vervolgens ook organisatiebreed (geaggregeerd en geanonimiseerd).

Goede verhalen inspireren ons. Dat maakt de kans groter dat we er iets mee gaan doen

10. Borg je inspanningen

Informatieveilig gedrag is een lijnverantwoordelijkheid. De inspanningen moeten erop gericht zijn, dat lijnmanagers die verantwoordelijkheid (kunnen) dragen. Alleen dan zorg je voor continue aandacht voor informatieveiligheid [22]. Betrek lijnmanagers bij de ontwikkeling van je programma. Vraag wat zij nodig hebben om hun informatieveiligheidstaak goed uit te kunnen voeren. Als zij het belang van het thema niet inzien, richt je dan eerst op hún bewustwording, al dan niet met behulp van de directie.

Ook dient iemand in de organisatie verantwoordelijk te zijn voor de continuïteit van het gehele programma. Diegene is 'eigenaar' van de interventies, checkt regelmatig of ze nog accuraat en aansprekend zijn en fungeert als aanspreekpunt voor de gehele organisatie. Meestal is dat de CISO of een soortgelijke functie.

Links

[1] 'Nationaal Cyber Security Beeld 5', NCSC: https://www.ncsc.nl/actueel/Cybersecuritybeeld+Nederland/cybersecuritybeeld-nederland-5.html

[2] https://nakedsecurity.sophos.com/2014/01/15/whos-to-blame-for-security-problems-surveys-say-everyone/

[3] 'Cyber security awareness campaigns: why do they fail to change behavior?', Global cyber security capacity centre 2014: https://www.sbs.ox.ac.uk/cybersecurity-capacity/system/files/Awareness%20CampaignsDraftWorkingPaper.pdf

[4] 'On the pulse. Information security risk in American business' Stroz Friedberg 2014: https://www.strozfriedberg.com/wp-content/uploads/2014/01/Stroz-Friedberg_On-the-Pulse_Information-Security-in-American-Business.pdf

[5] https://www.sbs.ox.ac.uk/cybersecurity-capacity/system/files/Awareness%20CampaignsDraftWorkingPaper.pdf

[6] 'Using behavioural insights to improve the public's use of cyber security best practices', Government office of science: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/309652/14-835-cyber-security-behavioural-insights.pdf

[7] https://www.incentro.com/nl/2016/02/16/90-van-werkend-nederland-omzeilt-eigen-it-afdeling/

[8] 'Hoe mensen keuzes maken. De psychologie van het beslissen', W.L. Tiemeijer. WRR, 2010: http://www.wrr.nl/fileadmin/nl/publicaties/PDF-overige_uitgaven/Hoe_mensen_keuzes_maken.pdf

[9] http://2012books.lardbucket.org/books/management-principles-v1.1/s06-personality-attitudes-and-work.html

[10] http://www.fractal.org/Bewustzijns-Besturings-Model/Spiral-dynamics.htm

[11] http://www.handhavingsacademie.info/wat-is-de-invloed-van-sociale-normen-op-gedrag/

[12] http://www.limburger.nl/cnt/dmf20160405_00015553/de-zaak-van-rey-iedereen-doet-het-en-iedereen-weet-het

[13] https://en.wikipedia.org/wiki/List_of_cognitive_biases

[14] https://en.wikipedia.org/wiki/Availability_heuristic#Vividness_effects

[15] http://blog.lelixxor.nl/2013/01/14/wetenschap/

[16] http://www.thestoryconnection.nl/Over-storytelling/Wat-is-storytelling-Wat-is-een-verhaal

[17] http://www.reneekeijzer.nl/de-zes-kenmerken-van-storytelling/

[18] 'Gedragsverandering via campagnes', Ministerie van Algemene Zaken, 2011: https://www.wageningenur.nl/nl/Publicatie-details.htm?publicationId=publication-way-343131303336

[19] Charles Jennings: 70-20-10: https://www.youtube.com/watch?v=t6WX11iqmg0

[20] 'Cybersecurity 2015 Awareness, gedrag & digitaal verantwoord ondernemen' NCTV 2015: https://www.nctv.nl/onderwerpen-a-z/rapporten.aspx?y=&m=&p5=rapport&p7=2006-01-01&select=3

[21] 'Human metrics, measuring behavior', SANS, securing the human: https://securingthehuman.sans.org/media/resources/presentations/STH-Presentation-HumanMetrics.pdf

[22] https://www.managementsite.nl/verbeter-cultuur-continu-verbeteren

WE LEKKEN EEN HALF JAAR; EN WAT HEBBEN WE GELEERD?

Een collega-ISO stuurt me een foto van de Telegraaf. Hij had zo ingeschat dat ik geen papieren krant zou lezen en wilde me (niet van harte) toch het nieuws laten zien. Ik lees "Vertrouwelijke gegevens van duizenden gehandicapte treinreizigers liggen op straat door laks privacybeleid…". Dat doet zeer.

Het is een datalek van de orde "beetje dom" want een sticker met UN+WW zichtbaar op een laptop die in het openbaar gebruikt wordt, is vragen om problemen. En natuurlijk reken ik dat mezelf ook aan. Het voelt best persoonlijk dat geroeptoeterde "laks privacybeleid" zelfs al weet ik dat het niet waar is en dat het ook nog eens om de Telegraaf gaat. Daarnaast schrijf ik zelf en weet ik dat het wel lekker bekt om berichten flink negatief en zwaar aan te zetten. Hoge bomen vangen nu eenmaal veel wind.

Hoe het ook voelt, het doet niets af aan de ernst van de zaak, het gaat immers om bijzondere gegevens en daar moet je bijkans nog voorzichtiger mee omspringen dan met gewone persoonsgegevens. Dat is dus falen. Overigens kan ik u één ding wel zeggen; de meldplicht heeft er wel voor gezorgd dat onze interne processen gesmeerd lopen en dat incidenten heel snel en goed worden opgepakt. Van onze fouten leren we nu hopelijk ook beter zodat we ze in de toekomst kunnen voorkomen.

Bij NS melden we trouwens bijna alles bij de Autoriteit Persoonsgegevens. Elk datalek is er namelijk één teveel. Alleen als we met zekerheid kunnen zeggen dat er geen impact is op de privacy, melden we niet. Klaarblijkelijk denkt niet iedereen hetzelfde, want de toezichthouder heeft medio mei 1.600 meldingen binnen en zegt dat dit cijfermatig niet kan kloppen. Het zouden er meer moeten zijn. Kijkend naar de eigen ervaring geef ik ze gelijk. Onderliggend issue komt met de mededeling van de Autoriteit echter niet naar boven. Want: hoe komt het nu dat er "te weinig" gemeld wordt? Ik denk dat een aantal zaken meespelen.

Ten eerste is het niet duidelijk wanneer een datalek al dan niet gemeld moet worden. Over iets ogenschijnlijk eenvoudigs als een verloren iPhone lopen de meningen immers al enorm uiteen. Ten tweede zijn we nog maar een paar maanden onderweg en het is daarom aannemelijk dat nog niet alle interne procedures vlekkeloos verlopen. Dat is overigens geen excuus maar wellicht wel een verklaring. Ten derde vinden sommige lekken hun weg niet naar de persoon die deze moet melden omdat degene met wetenschap van het lek simpelweg niet beseft dat het een lek is. Er zijn vast nog veel meer redenen te bedenken, maar dit zijn denk ik een aantal voor de hand liggende.

Het is nog een beetje vroeg in de wedstrijd om vergaande conclusies te trekken. Een voorzichtige tussentijdse conclusie is dat we met zijn allen veel meer lekken dan zichtbaar is of wordt. Waar ik vooral op hoop (naast wat meer guidance door de toezichthouder) is dat er iemand aan de slag gaat met een onderzoek naar die datalekken en daaruit niet alleen conclusies trekt maar ook werkbare aanbevelingen doet. Precies op dat moment denk ik dat we echt wat nuttigs hebben geleerd. En als ik u dan nog een klein trucje mag leren dat ik weer van een andere Privacy Officer heb geleerd: gebruik iMacros voor het melden van veelvoorkomende lekken. Eén keer een macro aanmaken en daarna bijna met één druk op de knop melden. Zo! Dat worden vast heel veel meer meldingen straks bij AP!

Mr. Rachel Marbus
@rachelmarbus op Twitter

TRENDS DIE KOPPIG ZIJN OF NIET LIJKEN TE BESTAAN

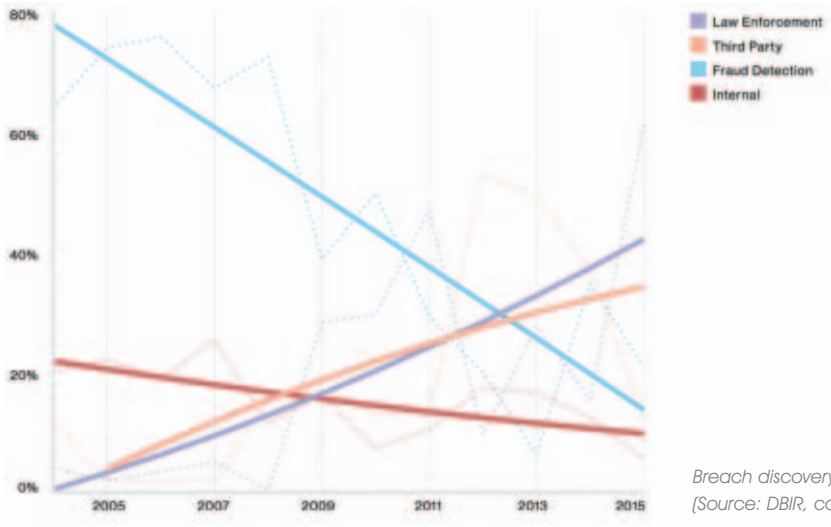
Verizon Data Breach rapport over 2015

Elk jaar stelt Verizon een omvangrijk rapport op over data breaches. Met deze term wordt zowel verlies als diefstal afgedekt. Het rapport beschrijft wat de trends zijn en doet dat op basis van de input van veel overheidsinstellingen zoals het NCSC, maar ook van commerciële partijen als Juniper.

Elk jaar weer wordt rapport met een klein mediaoffensief in de markt gezet. Voor Nederland betekent dat de mogelijkheid onder embargo vooraf de concept versie in te zien en daar met een van de auteurs over te praten. Het team kent het nodig verloop, dus eigenlijk is er elk jaar iemand anders die daarover uitleg geeft. Dit jaar was het Robinson Delaugerre (Lead Forensic Investigator), die vanuit Parijs antwoord gaf op alle vragen die we hadden, na het 85 pagina's tellende document te hebben doorgenomen. Het rapport is dit jaar toch weer anders van opzet. Het leest wat minder snel dan vorige edities. Er is wat de opzet betreft wel een pluspunt, het is makkelijker onderwerpen over te slaan die niet relevant zijn. Voor de financiële instellingen zijn de ontwikkelingen rond betaalterminals interessant, maar voor anderen minder. Dat hoofdstuk kun je dan overslaan zonder dat er verderop nog naar verwezen wordt.

De schrijvers slaan zelf ook bepaalde onderwerpen over. In de inleiding staat dat het IoT, in weerwil van alle horrorscenario's, nog nergens op betrapt is. Dat was de eerste vraag: "Kijken jullie wel naar alle mogelijke scenario's die aan IoT te linken zijn?".

Robinson: "Die vraag stellen we ons zelf ook, maar vooralsnog wijst alles erop dat IoT gewoon nog oninteressant is. Een C&C (command and control) server breng je veel makkelijker elders onder en er zijn doelwitten die bewezen makkelijker zijn". Via een korte verhandeling over het zwaar onderschatte risico dat printers en kopieermachines vormen, met de actuele case van



Breach discovery methods over time
(Source: DBIR, copyright Verizon)

gehackte printers in Duitsland als voorbeeld [1], kwam het gesprek weer terug bij de onderzoeksbevindingen en de volgende vraag: "Waarom zien we evenveel data breach-incidenten bij grote en kleine bedrijven in healthcare en information? Bij financials zie je juist dat de grote bedrijven eerder slachtoffer worden en bij retail zijn het juist de kleinere". Een verklaring daarvoor, anders dan dat voor deze twee sectoren duidelijk opgaat dat "size doesn't matter", is er niet. Nuttig is de bevinding wel: als je in deze sectoren werkzaam bent (of leverancier ervan bent) weet je dat je je als kleinere speler absoluut niet veiliger mag voelen, de cybercriminelen kijken niet alleen naar de grote spelers.

Wat voor de hostingsector goed te weten is: de onderzoekers hebben vastgesteld dat in 2015 de neergaande lijn doorzet dat servers de directe oorzaak van een data breach zijn. Servers an sich worden steeds veiliger en dat is goed nieuws. Nu het slechte nieuws. Het aantal data breaches waarbij personal devices (riep daar iemand BYOD?) als einddoel of als steppingstone voorkomt neemt explosief toe. Dat je via een lekke tablet uiteindelijk ook weer toegang tot een dichtgetimmerde server kunt krijgen spreekt voor zich. De onderzoekers zien dat ook voorkomen, maar wat er op de tablet zelf staat is in de regel al genoeg om een fors datalek te veroorzaken.

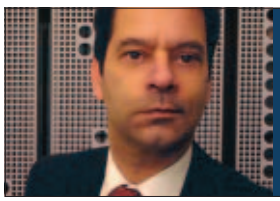
Een andere trend die de sector raakt: steeds vaker zijn het externe partijen die het datalek als eerste signaleren. Het lijkt er dus op dat externe monitoring echt werkt en de "onkunde" van partijen die de data lekken deel compenseren. Wat niet duidelijk werd, is hoeveel lekken dit soort externen voorkomen.

Laatste punt waar we uitgebreid bij stil stonden: inside jobs. Dat is, als het gaat om data breaches (vooral in de betekenis van opzettelijk lekken en ontvreemden), een verschijnsel waar weinig verbetering zichtbaar is. Het ontdekken ervan duurt lang en de makkelijkste manier om dat tegen te gaan wordt vaak over het hoofd gezien: continue controleren op onterechte privileges.

Voor inzichtelijke grafieken is het doorbladeren van het rapport voldoende. Voor meer details, ook over de boven genoemde trends, is het doorlezen van de betreffende hoofdstukken noodzakelijk. De eindversie van rapport is bij Verizon te downloaden [2].

Links

[1] Gehackte printers: http://www.heise.de/newsticker/meldung/Hackerangriff-Uni-Drucker-spuckten-Neonazi-Pamphlete-aus-3179949.html
[2] DBIR download: http://www.verizonenterprise.com/DBIR/



Rashid Niamat is journalist en werkzaam bij ISPam.
Rashid is te bereiken via rashid@niamatmediagroup.nl

TRENDSETTER De nieuwe DBIR

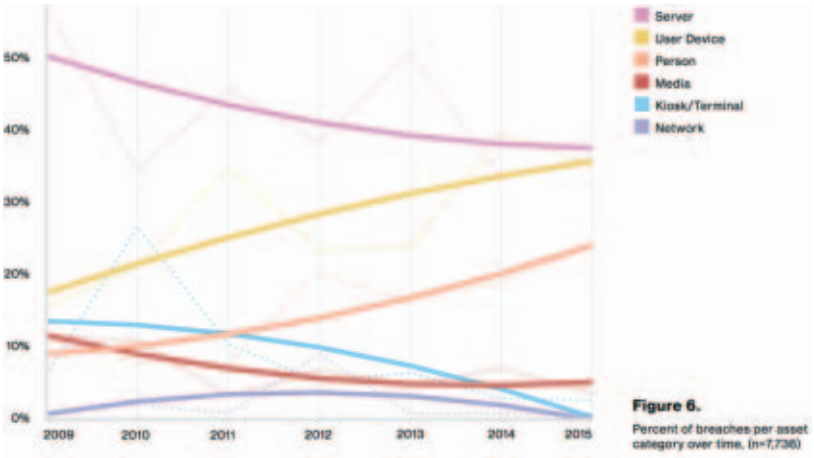


Figure 6.
Percent of breaches per asset category over time, (n=7,736)

Ik ben een fan van het Data Breach Investigations Report van Verizon. Zelf waarschuwen ze in de inleiding: "we would never suggest that every last security event of 2015 is in this report. We acknowledge sample bias, ..." Dit is nu juist waarom ik het rapport zo waardeer, die bescheidenheid. Al jaren, voor de negende keer, komt Verizon met de DBIR. En de resultaten zijn voor mij de beste weergave van de state of the union op het gebied van cybersecurity die er is.

Een aantal indrukken uit de overzichten, met de bijbehorende illustraties zoals in de DBIR genoemd:

- De fundamentele constatering blijft: de verdeling tussen externe/interne actoren is constant en ruim 80/20 (figure 2).
- De opkomst en de val van de cyberspionage. Die heeft heel duidelijk gepiekt in 2013 (figure 3). Maar lees wel mijn opmerking over misbruik van toegang door interen verderop.
- Hacking en Malware groeien exponentieel! Social engineering groeit mee (figure 4).
- Servers zijn nog steeds de meest aangevallen asset, maar ik verwacht dat dat volgend jaar niet meer het geval is: dan gaat het persoonlijk apparaat (BYOD!) de server overtreffen in data breaches. De data breach verschuift dus naar de eindgebruiker (figure 6), ook als figuur opgenomen bij dit artikel.
- Aantasting in minuten, extractie in dagen, met versnelling (figures 7, 8).
- En dan een verrassende: de effectiviteit van interne detectie (incl. fraude detectie) daalt, externe detectie neemt toe (figure 9). Ik heb hier geen goede verklaring voor. Wellicht is de sample bias hier debet aan: interne incidenten worden intern afgehandeld.

Nog wat andere opvallende zaken:

Phishing e-mails worden vooral in de eerste drie uur na ontvangst geopend (figure 14). En in 50% van die gevallen klikt de ontvanger ook op de link! Wat wordt er buit gemaakt? Dat is geen verrassing, credentials!

De aanvallen op web apps zijn een constante cocktail (figure 24). Aanvallen gebruiken bijna altijd deze combinatie:

- Gebruik van gestolen credentials;
- Gebruik van een backdoor of command & control server;
- Phishing;
- Spyware/keylogger;
- Command & control malware;
- Export van data.

De aanbeveling om two-factor authenticatie te gebruiken, input validatie voor alles en een alles-dekkend patchmanagement is een duidelijke. Daar valt niets op af te dingen.

Spionage viel terug, maar het is wel duidelijk dat er bij spionage op een heel andere manier gehandeld wordt: spionnen misbruiken in toenemende mate de interne medewerker en zijn toegangsrechten (figures 28, 29).

Al met al is er in mijn ervaring geen rapport dat zo direct bruikbaar is in de dagelijkse werkomgeving van een security officer, om richting management uit te leggen waarom er extra maatregelen nodig zijn (of niet), en om de juiste focus neer te leggen bij een project of een uitvoerende afdeling.

Door Lex Borger, hoofdredacteur IB magazine

INFORMED

An important aspect of good system design is that users should understand how the system works for their benefit. The attribute 'informed' is defined in the Big Blue Book of SABSA (Enterprise Security Architecture: A Business Driven Approach, Sherwood, Clark and Lynas) as follows:

"The user should be kept fully informed about services, operating procedures, operational schedules, planned outages, and so on."

The Attributer and The Attributer's Wife recently had a nice holiday. One of the visitor attractions that they experienced was to bathe in the hot waters of a famous thermal spring where, in a modern visitor centre, one can enjoy floating in the open-air pool of hot mineral water from deep in the earth, water that fell as rain 10,000 years ago. It is indeed a great thing to do on holiday. Why is this relevant? Because the facility has a 'system' for payment, entry and managing time slots.

At the entrance a nice young man explained that there is an entry fee for two hours of time. There is also a café that the couple planned to use for lunch. He told them that any time used in the café would be added to the two hours so that they could stay as long as needed in the café without wasting the time slot. He gave them each a 'smart wrist band' so they could charge any food and drink to the smart chip. He told them they should swipe the chip on entering the café and on exit (well, that's what they both heard). They were verbally given an exit time, to which any time in the café would be added.

Now, ask yourselves, readers, given this information, would you too make the following assumptions?

- Assumption 1: The smart chip records time of entry into the main facility and in and out of the café.
- Assumption 2: The smart chip also records the amount spent in the café that can be paid on exit from the main facility.

So they got their towels, dressing gowns and slippers, changed out of street clothes, and went to find the café. There they encountered a young lady whose English language skills were limited. She was also very dismissive and quite rude. She refused to scan the smart wristbands and said that she would scan them when she 'gave them their order' (she meant 'bill', they later learned). They waited a long time to be served and became quite agitated by this passing of time, because it seemed to them that their time was being wasted, their wrist bands not having been scanned. Some strong words were exchanged which did not lessen their confusion (yes, they had been misinformed) and they were not experiencing the calm relaxation promised.

Finally, another server, a man who spoke real English, attended them. He explained that the 'smart chips' had no knowledge of time (assumption 1) and only recorded the amount spent on food and drink (assumption 2). The server would tell them when they left how long they had been there and they could then add that on to their exit time. The time management is an entirely manual process and depends wholly on the trusted behaviour of the customer to be honest about when to leave (!!!). So, they had it half right. The Attributer had assumed (incorrectly) that this 'smart' system was fully automated and had a security level that would prevent fraudulent use of time.

Sometimes it pays to be dumb and not analyse everything, but if one is a systems engineer one tends to do it all the time. Systems are built from processes supported by technology, but the most important system components are the people – designers, operators and users. If the information given by some of the people to other people is wrong, then no technology can fix that. It's just a broken process. That is why in SABSA we take a 'total systems requirements approach' to systems engineering, and keep everyone correctly informed.

The Attributer

PUBLIC KEY PINNING OP EEN WEBSITE

Afgelopen jaren zijn er diverse problemen in het nieuws geweest die het vertrouwen in de Public Key Infrastructure (PKI) geschaad hebben en het wordt tijd om het vertrouwen weer terug te winnen. Maar het PKI-principe lijkt niet zo betrouwbaar meer omdat deze kan worden gehackt. Zie bijvoorbeeld Diginotar CA waarbij er zonder hun medeweten in hun naam certificaten konden worden uitgegeven die niet voldoen aan hun controles. Of dat leveranciers van hardware een Root CA's kunnen toevoegen aan een computer die niet aan de eisen van een volwaardige Root CA voldoen (zoals bijvoorbeeld bij SuperFish van Lenovo, Komodia en Dell). Hierdoor is het met malafide uitgegeven certificaten mogelijk een man-in-the-middle (MitM)-aanval uit te voeren zonder dat de gebruiker dat in de gaten heeft.

Naast dat het vertrouwen in certificaten aan het afbrokkelen is, is het ook mogelijk om een een TLS-sessie op een netwerk-node te onderbreken en de gegevens in te zien. Zakelijk wordt dat regelmatig gedaan om Data Leakage Prevention (DLP) in te richten. Daar wordt het uitgevoerd door systemen die 'SSL-Inspections' uitvoeren. SSL-Inspections is een techniek waar je de TLS-verbinding kunt onderbreken en bekijken, wat neerkomt op een MitM. Deze techniek is bruikbaar als je de werkstations legitiem volledig onder controle hebt, maar deze kan ook uitgevoerd worden met malware.
Ook deze manier van DLP zorgt voor verminderd vertrouwen in de PKI en daar is inmiddels een oplossing voor in de maak. Dit is een instelling op de webserver die de naam HTTP Public Key Pinning (HPKP) draagt.

Wat is HPKP?

HPKP is een methode om vanaf de server middels een hash-waarde aan de client door te geven welke certificaten geldig zijn voor de TLS-sessie. De hash-waarde wordt 'pin' (speld)

genoemd. Dit kan de Root CA zijn maar ook een Intermediate CA of het certificaat van de server zelf. Dat laatste is zeer af te raden omdat het nogal wat impact heeft op het beheer van de TLS-certificaten. Zolang de hash maar minimaal met één van de certificaten overeenkomt die in de Trust-chain van het TLS-certificaat zit.
Deze methode wordt in de PKI-wereld ook wel Certificate Pinning genoemd. Een techniek die al enkele jaren in SSH (Secure Shell) en mobiele apps van bijvoorbeeld banken gebruikt wordt om ervoor te zorgen dat de verbinding 100% te vertrouwen is. Deze nieuwe techniek hebben browser-fabrikanten (Mozilla en Google) opgepakt en beide bedrijven zijn deze aan het doorvoeren in hun browsers, Firefox en Chrome. Het HPKP-protocol is inmiddels een standaard.

De techniek

De webserver geeft een hash-waarde van een certificaat door aan de browser. De browser controleert dan of die waarde overheen komt met het certificaat dat hij heeft ontvangen. De browser slaat deze hash-waarde op zodat het bij een volgend bezoek aan de site de controle lokaal kan uitvoeren. Hiermee kan de browser bij nieuwe bezoeken een controle uitvoeren of deze waarde anders is dan bij eerste bezoek aan de website. Bij een MitM-aanval is er geen overeenkomst en weet de browser dat de verbinding niet goed is en verbreekt deze. Is het wel goed dan merkt de gebruiker niets en komt de verbinding tot stand.
Op de server moeten minimaal twee hash-waarden in de configuratie opgenomen worden om dit protocol te laten werken. Eén hash is van één van de certificaten uit de certificate-tree (Root CA of Intermediate CA) die gebruikt wordt en één hash is van een andere certificate-tree. Er kunnen meer hash-waarden opgegeven worden maar dat is niet noodzakelijk om het te laten werken. Beheerders moeten de configuratie van de webserver aanpassen als er een nieuw certificaat geïnstalleerd wordt. De hash-waarde verandert elke keer als het bijbehorende certificaat gewisseld wordt.

De configuratie

De hash-waarde wordt in een HTTP-header tijdens het opzetten van de TLS-sessie vanaf de server doorgegeven aan de browser. Deze waarde wordt via de configuratie van de webserver bepaald en hierdoor is het belangrijk deze op het juiste niveau te configureren. In Apache kun je de configuratie in het kader instellen en de hash-waarde kun je berekenen aan de hand van het gekozen certificaat: het Root CA-certificaat, Intermediate CA of het TLS-certificaat zelf.

Nadelen

Bij elke security-functionaliteit zijn er ook nadelen en soms zijn dat wel punten waar je goed over moet nadenken. HPKP is er één in die categorie, want als dat niet goed ingesteld is, werkt de bescherming niet of is de website niet meer toegankelijk. Men moet goed nadenken welke pinnen er in de configuratie opgenomen worden. Gebruik je de Root CA, Intermediate CA of het server-certificaat zelf. En er moet backup-pin opgenomen worden van een alternatieve CA. De verstandigste keuze is om de pinnen van de gebruikte Intermediate CA's op te nemen en deze goed bij te houden. Bij veranderingen van CA's moet de pin van tevoren aangepast worden. Doe je dat niet op tijd, dan kan het zijn dat bij een nieuw certificaat er mensen niet meer op de website kunnen komen.

Conclusie

Het is nog niet optimaal want ondersteuning in browsers loopt nog achter. Zover ik nu weet zijn alleen Chrome en Firefox voorzien van deze functie. Naast ondersteuning in browsers is een stukje configuratie aan serverzijde noodzakelijk. En men moet goed in de gaten houden dat als het certificaat aan serverzijde verloopt en aangepast wordt, er een nieuw configuratie op de server doorgevoerd moet worden.
Dit protocol (HPKP) in combinatie met Perfect Forward Secrecy (PFS) is een welkome toevoeging op de veiligheid tegen MitM-aanvallen.

Harld Röling is werkzaam bij de afdeling CISO van een Nederlandse bank en heeft zich vanaf 1999 gespecialiseerd in de technische en organisatorische aspecten van asymmetrisch Key Management en PKI. Daarnaast is Harld vrijwilliger bij Bits of Freedom, bij Privacy Cafe's en de Toolbox. Harld is bereikbaar op e-mailadres harld.roling@hroling.nl.

OPEN SOURCE VIDEOSURVEILLANCESOFTWARE

In het lage segment van de beveiligingsmarkt is de prijzenslag begonnen. Voor een paar honderd euro heb je tegenwoordig een camerasysteem. Uit recente meldingen blijkt dat er toch vaak dubieuze software in dit soort systemen zit. Zo werd eind 2015 een achterdeur ontdekt in de firmware van een heel goedkoop videosurveillancesysteem van het merk MVPower [1]. Via deze achterdeur werden screenshots gemaakt maakt en doorgestuurd naar een e-mailadres in China. Als u op zoek bent naar een goedkoop videosurveillancesysteem en toch grip wilt krijgen op de veiligheid van de software kan een opensourcesysteem een flexibel alternatief bieden.

Open source betekent letterlijk eigenlijk 'open broncode', wat inhoudt dat de broncode van de software vrij mag worden gebruikt, ingezien en aangepast. Het is dus door ieder in te zien en te gebruiken. Veel opensourceproducten hebben een groot netwerk van ontwikkelaars en gebruikers om zich heen. Leden van deze netwerken kunnen voorstellen doen om de bestaande code te veranderen of te verbeteren. Hierdoor worden fouten in de beveiliging snel ontdekt en gedicht en worden de beveiligingsupdates snel gedistribueerd. Het resultaat is een veilig stuk software, zowel nu als in de toekomst.

Zoneminder

Het videosurveillancesysteem Zoneminder is gebaseerd op opensourcesoftware en het Linux-operating-system. Naast de Zoneminder-software zijn ook de opensourceproducten MySQL en PHP benodigd. Tenslotte dient de Linux-server voorzien te zijn van een webserver, zoals bijvoorbeeld Apache en dienen tools zoals C++, Perl aanwezig te zijn.
Zoneminder is opgebouwd uit verschillende bouwstenen. Eén van die bouwstenen is het populaire Video4Linux (V4L)-framework. Video4Linux is een verzameling van device-drivers en API's voor het ondersteunen van real-time video-opslag op

Linux-systemen. Video4Linux ondersteunt veel soorten camera's op zowel IP- als USB-technologie. De output van Video4Linux is gestandaardiseerd en kan daardoor eenvoudig door programmeurs en applicatiebouwers te gebruiken.
Zoneminder heeft een zeer complete web-based-interface en ondersteund zelfs Android- en iOS-apps en is daardoor via internet toegankelijk en van overal in de wereld. De web-interface is hierdoor ook geschikt voor 'headless' systemen. Een headless systeem is een computer zonder beeldscherm, grafische gebruikersinterface (GUI) of een toetsenbord en muis. Zoneminder is daarom ook geschikt voor embedded systemen. Ook is er een eenvoudige xHTML-interface beschikbaar waarmee elementaire besturing mogelijk is, vanuit de wat oudere mobiel telefoons, op basis van deze xHTML.

Veelzijdig

Dit videosurveillancesysteem kan worden gebruikt in een omgeving waar meerdere camera's gewenst zijn. Het is schaalbaar en dus geschikt voor thuisgebruik en klein zakelijk gebruik maar ook voor multi-server-enterprise-toepassingen is het bruikbaar. De software is compatibel met alles van de Raspberry Pi tot de meest moderne serverhardware. Groot gebruiksvoordeel is dat door het gebruik van open source software een vendor lock-in wordt voorkomen en is men niet gebonden aan dure licenties of apparatuur.
Naast de elementaire systeemfuncties voor videosurveillance ondersteunt Zoneminder aanvullende functionaliteiten. Zo zijn er bovenop elementaire functies als capture, analyse, recording en het monitoring van videobeelden handige functies waarmee u kunt pauzeren, terugspoelen en zelfs functies als digitale zoom zowel live en historische video zijn aanwezig. Ook ondersteund Zoneminder camerabesturing zodat gebruikers camera's kunnen draaien en bewegen. Hiervoor worden de verschillende protocollen voor de pan/tilt/zoom-camera's ondersteund.
Een ander probleem bij "normale" videosurveillancesystemen is ondersteuning van camera-drivers. Zoneminder ondersteunt

de software is compatibel van de Raspberry Pi tot moderne serverhardware

een breed spectrum van camera's van de verschillende merken en technologieën.
Net als in de commerciële systemen heeft Zoneminder de functie om events te registeren en schijfruimte te besparen. De Zoneminder-geeks noemen deze functie " Modect" het geen staat voor MOtion DeteCTtion. Met Modect worden alleen de bijbehorende beelden opgenomen wanneer de camera beweging detecteert.
Wat betreft communicatiemogelijkheden ondersteunt Zoneminder naast IP-technologie zelfs de X10 netwerken. X10 is een open standaard voor een communicatieprotocol dat gebruik maakt van het bestaande stroomnet. Een populaire naam voor X10 is "Powerline". Camera's zijn vrij eenvoudig via Powerline aan te sluiten en doordat er gebruik wordt gemaakt van het lichtnet scheelt dit een hoop aan bekabeling.

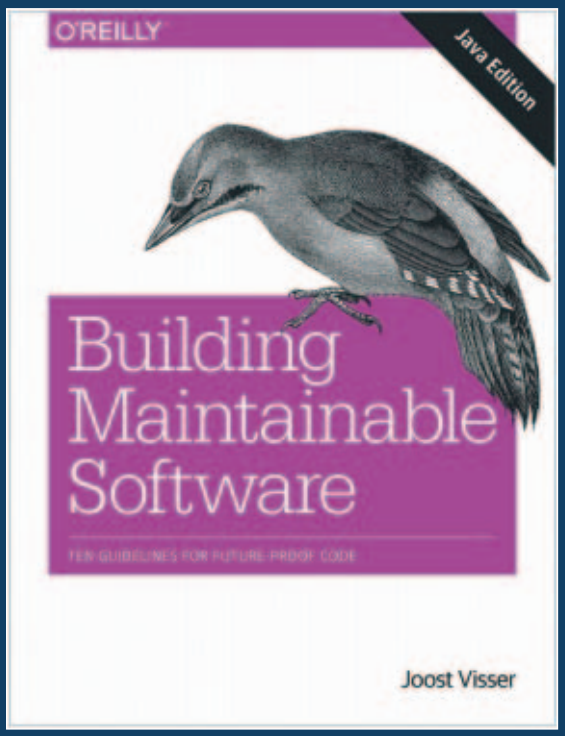
Conclusie

Zoneminder is open-source-videosurveillancesoftware, met een grote groep enthousiaste ontwikkelaars en gebruikers. Op internet zijn verschillende ZoneMinder-forums te vinden. Vanaf de website [2] kan men voor de verschillende operating-systems de software downloaden. Zo zijn er softwarebundels voor Ubuntu, RedHat, Debian, Gentoo. Kortom met Zoneminder is relatief eenvoudig een leuk en veilig videosurveillancesysteem in elkaar te zetten.

Links

[1] Backdoor in MVPower: http://news.softpedia.com/news/backdoor-in-mvpower-dvr-firmware-sends-cctv-stills-to-an-email-address-in-china-500502.shtml
[2] Zoneminder site: https://zoneminder.com

Ronald Eygendaal schrijft sinds 1990 over informatiebeveiliging, elektronische & technische beveiliging, fraudedetectie & -bestrijding, en bewaking & beveiliging voor de toonaangevende vakbladen in Nederland en België. Ronald is bereikbaar via ronald.eygendaal@kpn.com.



Titel: Building Maintainable Software
Subtitel: Ten Guidelines for Future-Proof Code
Auteurs: Joost Visser, Sylvan Rigal, Rob van der Leek, Pascal van Eck, Gijs Wijnholds
Taal: Engels
Pagina's: 164 (xix+145)
Uitgever: O'Reilly Media
Datum: December 2015
ISBN: 978-1-491-94434-9

ONDERHOUDSVRIENDELIJKE SOFTWARE

Auteur: Lex Borger, Hoofdredacteur van IB magazine

Ik kreeg het boek 'Building Maintainable Software' van Joost Visser e.a. ter review. Ik vond het een goed boek om te lezen, maar is het een boek dat een review waard is in dit blad? Daar moest ik even goed over nadenken. Ik heb hier zelf een vooroordeel over: ik ben de informatiebeveiliging ingerold via de softwarekwaliteit.

Lang hoefde ik er niet over na te denken, want het antwoord staat eigenlijk al op pagina twee in het boek. Hier worden de vier typen van onderhoud genoemd:

- Het ontdekken en repareren van bugs (updates tijdig uitvoeren is altijd nodig);
- Het aanpassen aan de omgeving (zoals automatisch gebruik maken van aangesloten hardware of nieuwe software);
- Het toevoegen van nieuwe functionaliteit;
- Het robuuster maken (verhogen van kwaliteit, voorkomen van problemen).

Twee van de vier hebben een directe link met informatiebeveiliging. Welke niet? Het aanpassen aan de omgeving en het toevoegen van nieuwe functionaliteit – tenzij het natuurlijk om beveiligingsfunctionaliteit gaat. Dus er is minstens altijd een indirecte link.

Vervolgens legt Joost uit waarom 'maintainability' belangrijk is. Onderhoudbare software is veel makkelijker aan te passen dan niet-onderhoudbare software. Het boek stelt dat het verschil een factor twee is. Ik herken dat uit mijn ontwikkelingsverleden. Wij wisten heel goed welke producen onderhoudbaar waren en welke niet. Als je opgezadeld werd met een issue in zo'n probleemgebied, dan kreeg je vanzelf al meer ruimte van je collega's. Als je in zo'n product problemen kon oplossen kreeg je respect van je collega's.

De rest van het boek is vrij eenvoudig van structuur: er worden

drie principes geformuleerd, er wordt uitgelegd hoe onderhoudbaarheid meetbaar gemaakt kan worden en er worden tien eenvoudige richtlijnen gegeven, één richtlijn per hoofdstuk.

De drie principes zijn:

- Onderhoudbaarheid verkrijg je door het volgen van eenvoudige richtlijnen;
- Onderhoudbaarheid breng je niet achteraf aan, iedereen doet het de hele tijd;
- Sommige overtredingen zijn erger dan anderen.

Om deze principes goed te begrijpen helpt het ook de richtlijnen te kennen:

1. Houd je code-eenheden kort
2. Houd je code-eenheden eenvoudig
3. Schrijf code maar één keer
4. Houd interfaces van eenheden klein
5. Splits belangen in modules
6. Koppel componenten losjes
7. Houd je architectuur-componenten in balans
8. Houd je softwarepakketten klein
9. Automatiseer je ontwikkeling en testing
10. Schrijf nette code

De vertaling is voor mijn rekening, ik realiseer me dat programmeren eigenlijk een heel Engelse aangelegenheid is. Toch vond ik het in het kader van deze review belangrijk om het ook voor niet-programmeurs toegankelijk te maken.

Elk hoofdstuk legt de richtlijn heel praktisch uit, met voorbeelden van slechte en goede programmaregels in Java. Het boek heet ook de 'Java-Edition'. Er wordt ook gewerkt aan een C++-versie. Dit is een probleem voor elk praktisch boek over programmeren: welke programmeertaal gebruik je voor je voorbeelden?

Donald Knuth gebruikte in zijn epos 'The Art of Computer Programming' de fictieve taal MIX voor de fictieve 1009 computer. In die dagen hadden talen namen en computers nummers (voor oplettende lezers: wat is de waarde van MIX in Romeinse cijfers?). Ik weet niet of Joost dacht met alleen een Java-versie van het boek weg te komen, maar de verschillende kampen van programmeurs zijn kennelijk al in opstand gekomen. Waar je in programmeert, is je religie.

In ieder geval is het functioneel niet storend dat de voorbeelden in Java zijn. De essenties worden goed uitgelegd en de voorbeelden zijn eenvoudig. Een niet-programmeur met wat affiniteit voor logica zou het kunnen volgen. Ik heb nooit in Java geprogrammeerd, maar had er totaal geen moeite mee de voorbeelden bij eerste lezing te volgen.

Van elke richtlijn wordt het praktisch nut bewezen met meetgegevens uit het archief van de Software Improvement Group (SIG). Daarmee wordt inzichtelijk hoe de richtlijnen in de praktijk werken.

Een andere praktische inslag is het presenteren van mogelijke tegenwerpingen op de richtlijnen. Het boek bevat dat bij elke richtlijn. Het geeft je het gevoel de discussie mee te maken en daarmee ben je ook beter voorbereid op dat soort discussies op de werkvloer.

Het mooie is dat de richtlijnen echt eenvoudig en duidelijk zijn. Onafhankelijk van de taal kun je spreken over het aantal parameters van een code-eenheid. Wel moet je wat abstractie toepassen: een code-eenheid (unit of code) kan in een taal een procedure heten, of functie, routine, en vast nog een paar namen die ik niet meer weet. En in bepaalde talen heeft 'unit' een specifiek andere betekenis. Daar moet je wel doorheen kunnen lezen.

Als ik terugdenk aan mijn verleden toen ik nog programmeerde, dan kan ik alle richtlijnen onderschrijven. Er zijn richtlijnen die heel eenvoudig lijken, maar dat niet zijn, bijvoorbeeld 'schrijf code maar één keer'. Het is soms echt hard werk om bij het coderen zeker te maken dat je code zoveel mogelijk hergebruikt. Je moet zoeken naar de juiste code, je moet zorgen dat je de code dan invoegt in je project. Is de code wel meteen bruikbaar? Wellicht moet het eerst aangepast worden - omdat de routine te veel doet. Misschien zit de code op een onhandige plaats. Om iets geschikt te maken voor hergebruik ben je soms flink bestaande code aan het aanpassen (ook wel refactoren genoemd). En een kleine aanpassing wordt een grote...

Als je dit moet doen in code waar je al minder bekend mee bent, dan wil je hier nog wel eens tegen in verzet komen. Als je als programmeur verantwoordelijk bent voor een module in het OS en je past een compiler aan als onderdeel van een nieuwe functionaliteit, dan zit je niet te wachten op review-commentaar dat eist dat je nog wat refactoring doet aan de compiler. Toch was dat, gezien de richtlijnen in dit boek, toen de juiste beslissing.

Conclusie: het is een goed en vooral praktisch boek. Ik kan het iedere programmeur en tester aanraden, en ook iedere manager onder wiens verantwoordelijkheid software ontwikkeld wordt.

Links

[1] Boek op de website van SIG: https://www.sig.eu/en/building-maintainable-software

[2] Boek op de website van O'Reilly: http://shop.oreilly.com/product/0636920049159.do

Eric Luijff neemt de honneurs waar voor de twee dames en neemt de eerste prijs in ontvangst van Renato Kuiper.

PRIJSUITREIKING ARTIKEL VAN HET JAAR 2015

Methodisch grondig aangepakt en uitgewerkt

Op een zonnige avond in Veenendaal werden op 19 april traditiegetrouw tussen de ALV en de avondmeeting de prijzen uitgereikt voor Artikel van het Jaar. Renato Kuiper, als voorzitter van de jury en oud-hoofdredacteur nam de honneurs waar en presenteerde het juryrapport. Aan de auteurs die aanwezig waren werden de prijzen persoonlijk uitgereikt. Inmiddels hebben alle auteurs hun prijzen ontvangen.

Renato vertelde hoe de jury te werk was gegaan met de voorselectie van de redactie. Dit jaar waren twee prijzen toegekend aan artikelen met meerdere auteurs, wat het aantal winnende auteurs op vijf zet.

Ervaring is kennelijk niet nodig om te winnen: drie van de vijf auteurs wonnen met hun debuut in het blad. Eén van de doelstellingen achter de prijs is het aantrekken en aansporen van nieuwe auteurs. Zonder te kunnen zeggen of er een causaal verband is, is de redactie wel blij met deze ontwikkeling.

De redactie is blij om te zien dat het aantal vrouwelijke winnaars in één klap verdubbeld is.

Na het recente fiasco van V&D-cadeaubonnen, waarbij miljoenen aan gekochte bonnen zonder waarschuwing ineens ongeldig gemaakt werden, had de redactie nog overwogen de prijzen te veranderen. Uiteindelijk heeft de redactie geen verandering doorgevoerd. De prijzen zijn respectievelijk € 500,- € 100,- en € 100,- voor de eerste, tweede en derde prijs, uitgekeerd in Bol.com-bonnen.

Derde prijs

De derde prijs is toegekend aan Henk-Jan van der Molen. Hij won vier jaar geleden ook al een prijs. De jury beloonde zijn artikel 'Veiliger met voorkennis' omdat hij hierin verschillende invalshoeken liet samenkomen.

Tweede prijs

De tweede prijs was voor Martijn de Hamer en Don Stikvoort. Beiden waren niet aanwezig. Hun artikel 'CSIRT Maturity Kit' toonde een holistische kijk op het thema incident response.

Eerste prijs

De eerste prijs was voor twee dames, Milena Janec en Eldine Verweij, die hun tweeluik 'Advanced Business Impact Analyse' volgens de jury methodisch grondig hadden aangepakt en uitgewerkt. Beiden waren niet aanwezig, maar een oude bekende van de vereniging, Eric Luijf, nam de prijs voor hen in ontvangst.

De redactie is blij om te zien dat het aantal vrouwelijke winnaars in één klap verdubbeld is. Informatiebeveiliging is bij uitstek een vakgebied dat door de veelzijdigheid gebaat is bij zoveel mogelijk diversiteit. We zijn met nummer 4 inmiddels nog maar halverwege 2016, dus er is nog gelegenheid genoeg voor auteurs om mee te dingen naar 'Artikel van het Jaar 2016'.

Achter Het Nieuws

In deze rubriek geven enkele IB-redacteuren in een kort stukje hun reactie op recente nieuwsitems over de informatiebeveiliging. Dit zijn persoonlijke reacties van de auteurs en deze geven niet noodzakelijkerwijs het officiële standpunt weer van hun werkgever of van PvIB. Vragen en/of opmerkingen kunt u sturen naar ibmagazine@pvib.nl

KWETSBAARHEID OFFICE 365

Onlangs vonden hackers een kwetsbaarheid in de SAML-implementatie van Office 365. De kwetsbaarheid maakte het mogelijk om toegang te krijgen tot de accounts en data van bedrijven die met hun eigen active directory (federated) op Office 365 zijn aangesloten. Al verhielp Microsoft het probleem razendsnel: het was niet de eerste keer dat de data van gebruikers door een authenticatiefout gevaar liep. Welke nieuwe kwetsbaarheden brengt een cloud-oplossing als Office 365 eigenlijk met zich mee? En wat lost het op beveiligingsgebied op? Onze redacteuren geven hun mening.

Lex Dunn

De kwetsbaarheden van cloud-applicaties (SAAS: software as a service) zijn op zich natuurlijk niet nieuw, het nieuwe is dat het niet meer in jouw domein zit, maar in het domein van een "ander". En je moet er dus maar op vertrouwen dat die "ander" er net zo zorgvuldig mee om gaat als je zelf zou doen. Omdat het gaat om generieke applicaties, waar het verdienmodel in het volume zit, zal zo'n cloud-provider echt niet voor jou alleen aan de slag gaan. Je kunt proberen dat via een contract af te dwingen, maar je zult merken dat ze alleen hun eigen "algemene voorwaarden" willen hanteren. Ze schermen dan vaak met ISO 27001-certificatie of een ISAE 3000-rapport, maar als je niet weet wat de basis daarvoor is, heb je er weinig aan. Bovendien ben ik in het verleden "kleine lettertjes" in zulke voorwaarden tegengekomen, die stellen dat de provider het recht heeft om alles (inclusief JOUW data) aan een ander bedrijf te verkopen… Kortom, je zult jezelf wat beter moeten beschermen als je cloud-applicaties gaat gebruiken. Maar dat kan lastig worden: het meest gebruikte advies bij opslag in de cloud is om alles zelf te versleutelen. Dat gaat natuurlijk wel op bij opslag (file share) in de cloud, maar bij de meeste cloud-applicaties (denk aan


Lex Dunn


Lex Borger


Maarten Hartsuijker

bijvoorbeeld ziekteverzuim, CRM, boekhouding) zal dat niet mogelijk zijn omdat ze standaardformulieren (met datacontrole) gebruiken. Dus overweeg zorgvuldig of je die vertrouwelijke bedrijfsgegevens, of gegevens van jouw klanten, wel op die manier in de cloud wilt brengen.

Maarten Hartsuijker

De risico's die je loopt of verhelpt met een cloud-oplossing zijn heel erg organisatie-afhankelijk. Voor kleinere organisaties is het vaak lastig om beschikbaarheids- en continuïteitswensen volledig in-huis te organiseren. Cloud-oplossingen maken schaalbare IT-oplossingen voor iedereen toegankelijk. Ook zullen veel grotere cloud-organisaties informatiebeveiliging veelal beter geregeld hebben dan kleine organisaties. Voor hen is het lastig om hier specifiek aandacht aan te besteden. Voor grotere organisaties is het makkelijker om zelf een betrouwbare IT-omgeving in te richten. Een omgeving waarin ze beperkt afhankelijk zijn van andere partijen en geen risico's erven van kwetsbaarheden die andere cloud-klanten al dan niet bewust introduceren. Of die de cloud-leverancier zelf introduceert in zijn poging om op één platform meerdere klanten te faciliteren. Door de schaalgrootte van de eigen organisatie heb je de cloud niet per definitie nodig om kostenefficiënt te zijn. Het type kwetsbaarheden van een cloud verschilt enorm van het type kwetsbaarheden binnen een organisatie-specifieke IT-omgeving. Elke beveiliger weet dat de hoeveelheid functionaliteit die je beschikbaar stelt in grote mate bijdraagt aan het aanvalsoppervlak en de mogelijk aanwezige kwetsbaarheden. En een organisatie waar je eerst met een token door een poort heen moet voor je bij gedetailleerde applicatieve functies terecht komt, heeft daardoor een veel beperkter aanvalsoppervlak dan een cloud-leverancier die "shared" een hele applicatie beschikbaar stelt aan iedereen die ervoor wil betalen.
Een fout zoals in de SAML-implementatie van Office 365 naar voren kwam, treft alleen cloud-klanten van Microsoft, omdat het een fout is die voortkomt uit het feit dat deze cloud meerdere klanten tegelijk moet bedienen. Het is mede omwille van dit soort fouten en het brede aanvalsoppervlak binnen een cloud dat bijvoorbeeld de Amerikaanse overheid aan Microsoft heeft gevraagd om overheidsdiensten te bedienen vanuit een aparte cloud. Hierin zitten logische en fysieke scheidingen die borgen dat kwetsbaarheden in de commerciële cloud geen impact op de overheid hebben. Voor veel kwetsbaarheden biedt het compartimenteren van IT-omgevingen uiteindelijk meer bescherming dan de applicatieve scheiding die in veel clouds gebruikelijk is.

Lex Borger

De basis van deze kwetsbaarheid ligt bij het gegeven dat één partij de authenticatie uitvoert en een andere partij de dienst levert. Een clouddienst die afgenomen wordt door een enterprise werkt typisch op deze manier, maar er zijn ook andere scenario's die authenticatie en autorisatie splitsen, bijvoorbeeld omdat authenticatie juist geïsoleerd wordt geïmplementeerd van een businessapplicatie. Isolatie van belangrijke diensten is een goed architectuurpatroon, of het nu gaat om interne automatisering of extern zichtbare ('internet-facing') automatisering. Het staat toe dat we impact isoleren, risico's spreiden, maatregelen kunnen focussen en complexiteit verlagen. Allemaal zaken die goed zijn voor de beveiliging. Er komt wel wat bij: deze componenten moeten verbonden worden en elkaar kunnen vertrouwen. Vertrouwde verbindingen tussen verschillende diensten gebruiken in de context van een web-applicatie is niet makkelijk. Je moet je als afhankelijke partij constant afvragen of je met de juiste dienst verbonden bent. Op het internet hebben we hier TLS voor, als de verbinding direct is. Maar juist in dit geval is dat niet zo. Authenticatie wordt via een re-direct in de browser van de klant gedaan. Dus het mechanisme is in dit geval anders.
De klant authenticeert zich bij de authenticatiedienst en komt bij de afhankelijke dienst terug met een verslag daarvan, de SAML-assertion. Als het goed is (en met SAML 2.0 is dat zo), dan is de assertion cryptografisch getekend en dus te controleren. Een medewerker van een bedrijf met Office 365 die zich meldt bij Microsoft wordt naar de authenticatiedienst van haar bedrijf gestuurd, alwaar zij op vertrouwde manier kan inloggen. Microsoft ontvangt de SAML-assertion, checkt deze, bepaalt wie zij is uit de assertion en zet een Office 365-sessie op. De medewerker is nu ingelogd.
Microsoft voerde de checkt niet goed uit. Het goede nieuws is: deze check is niet ingewikkeld. Ze hadden slechts zeven uur nodig om de check in productie te krijgen. Het slechte nieuws is: de potentiële impact is enorm! Eén kwaadaardige hacker kan zich voordoen als een willekeurige andere bekende klant. Maar dit kan niet ongestraft, want deze maskerade is wel in de logs van Microsoft terug te vinden. Ik geloof het daarom meteen als ze stellig kunnen zeggen dat er geen misbruik is geweest.
Moraal is: als je een dienst aanbiedt op het internet, check, check en dubbelcheck! Elke interactiestap met een andere dienst kan potentieel gemanipuleerd zijn. En doe dit zeker met de diensten waar je fundamenteel je beveiliging op bouwt.

Links

[1] Blog van Ioannis Kakavas: http://www.economyofmechanism.com/office365-authbypass.html
[2] Blog van Klemen Bratec: https://bratec.si/security/2016/04/27/road-to-hell-paved-with-saml-assertions.html

COLOFON

IB is het huisorgaan van het Platform voor InformatieBeveiliging (PvIB) en bevat ontwikkelingen en achtergronden over onderwerpen op het gebied van informatiebeveiliging.

PvIB
Platform voor InformatieBeveiliging

REDACTIE
Lex Borger (hoofdredacteur, werkzaam bij i-to-i)
e-mail: hr@pvib.nl
MOS bv, Nijkerk (eindredactie)
e-mail: ibmagazine@pvib.nl

Tom Bakker (Digidentity BV)
Kas Clark (NCSC)
Lex Dunn (Capgemini)
Maarten Hartsuijker (Classity)
Rachel Marbus (NS)
Bart van Staveren

ADVERTENTIE-ACQUISITIE
e-mail: adverteren@pvib.nl;
of neem contact op met MOS
T (033) 247 34 00
ibmagazine@pvib.nl

VORMGEVING EN DRUK
VdR druk & print, Nijkerk
www.vdr.nl

UITGEVER
Platform voor InformatieBeveiliging (PvIB)
Postbus 1058
3860 BB NIJKERK
T (033) 247 34 92
F (033) 246 04 70
e-mail: secretariaat@pvib.nl
website: www.pvib.nl

ABONNEMENTEN 2016
De abonnementsprijs in 2016 bedraagt € 118,50 (exclusief btw), prijswijzigingen voorbehouden.

PvIB abonnementenadministratie
Platform voor InformatieBeveiliging (PvIB)
Postbus 1058
3860 BB NIJKERK
e-mail: secretariaat@pvib.nl

ISSN 1569-1063

SPANNEND

Het hacken van omgevingen zal voor veel mensen een hele uitdaging blijven, en bij wie het doet zal het toch altijd weer spanning oproepen.

Ik herinner mij de tijd dat ik nog geen grijs haar had, nog geen hypotheek en nog geen kabelmodem. In de tijd waarover ik spreek deden we alles nog met een 'normaal' modem. Ik had eigenlijk al best een moderne, want met zijn 14k4 (2400 baud) was hij de meeste modems nog wel de baas. Met spanning zat je achter je PC en kon je 'wardialen' (lukraak proberen een nummer te bereiken waarvan je hoopte dat er een dataverbinding achter hing). Vervolgens kon je een beetje neuzen op de netwerken van bedrijven. Vaak kon je op de zogenaamde bulletinboards wel wat beginstukjes vinden van je puzzel en zat je met bonzend hart te kijken wat er allemaal te vinden was. Gevolg? Hoge telefoonrekeningen, avonden achter elkaar niet te bereiken en een partner die je moest uitleggen waarom de telefoonrekening zo hoog was.

Prachtig! En ik denk daar met veel plezier aan terug. Ik heb weleens wat gevonden en wat doorgelezen maar alles netjes laten staan en niets met de informatie gedaan, zoals het hoort. Wij noemden dat in die tijd hacken, maar dat is niet meer te vergelijken met de hackers van tegenwoordig. Die halen momenteel een flinke buit weg… Natuurlijk is het internet gekomen, waardoor veel meer gegevens gekoppeld zijn en te vinden zijn. Hackers zijn er tegenwoordig in alle soorten en maten. Een tienjarig jongetje die de beveiliging van Instagram kraakt en gegevens weet te stelen. Hij kreeg een beloning van tienduizend dollar. Een schijntje als je nagaat dat Facebook, de eigenaar van Instagram, jaarlijks ruim één miljoen aan bug-bounties uitkeert. Een hacker uit Rusland die ruim tweehonderdzeventigmiljoen userid's en passwords heeft gestolen van emailaccounts van onder andere Gmail en Hotmail. De hacker vraagt slechts zestig roebel (zesenzestig cent) voor de hele verzameling. Hij wil geen geld verdienen, hij wil slechts dat de koper hem ophemelt op hacker-forums. Het gaat hem dus alleen om de roem.

Waarom haal ik juist deze twee voorbeelden aan?

Het eerste voorbeeld omdat blijkt dat grote bedrijven niet meer in staat zijn een garantie te geven over de veiligheid van hun data. Daarom nodigen ze iedereen uit om de gaten in hun omgeving bloot te leggen en het hen te melden tegen een beloning. Het feit dat er één miljoen per jaar uitgekeerd wordt en de beloning van het jongetje slecht tienduizend dollar was, betekent dat er dus honderd lekken gevonden zijn, oftewel twee per week. Wij spreken dan over Facebook, de eigenaar van onder andere Instagram en Whatsapp.

Het tweede voorbeeld betreft de emailgegevens van onder andere Gmail, oftewel Google. De handel van Google bestaat uit het aanbieden van ruimte om data te plaatsen en je email te faciliteren. Vervolgens verzamelen ze met allerlei programma's veel data van de gebruiker, die vervolgens verkocht worden aan adverteerders. Ik begrijp niet dat het mogelijk is om inloggegevens van zoveel gebruikers te hacken. Inmiddels lijkt het erop dat veel van de gestolen data niet klopt en dat het verhaal niet waar is, maar we wachten dat gewoon even af.

Berry

SECO INSTITUTE

wereldwijd
PARTNER-
NETWERK
Trainingsinstituten

EXIN
Examineren

Cyber Security & Governance
Certificeringsprogramma

Het **Security & Continuity (SECO) Institute** presenteert het Cyber Security & Governance certificerings-programma. Dit internationale ceritificeringsprogramma is ontwikkeld door opleider Security Academy en exameninstituut EXIN.

Het certificeringsprogramma is wereldwijd beschikbaar via door SECO geaccrediteerde opleiders. Er worden 7 opleidingstracks aangeboden op het gebied van Information Security, IT-Security, Privacy & Data Protection, Ethical Hacking, Secure Software, Business Continuity en Crisis Management. Iedere track bestaat uit 4 certifi-ceringslevels (voor de beginner t/m. de gevorderde) namelijk:
Foundation, Practitioner, Expert en Certified Officer.

Beschikt u al over een certificering zoals CISSP, CISM, C/CISO, zet dan de volgende stap in uw carrière met een van onze Expert opleidingen. Voor meer informatie zie www.seco-institute.org.

www.seco-institute.org info@seco-institute.org +31(0)348-408061