Zusammenfassung - NASS

SK

27. Februar 2016

Korrektheit und Vollständigkeit der Informationen wird nicht gewährleistet.

Inhaltsverzeichnis

1 Introduction

1.1 Taxonomie der Angreifer

- einzelner Angreifer
  - sozialer Hintergrund
  - öffentliche Aufmerksamkeit als Antrieb
  - evtl pol. Statements

  - geht gewöhnlich niedrige Risiken ein
- organisierte Kriminalität
  - Geld als Antrieb
  - mittlere Risiken
- Terroristen
  - politische oder gesellschaftliche Motivation
  - hohe Risiken
  - Zerstörung/Verwirrung als Ziel
- Konkurrenten
  - möglichst niedriges Risiko der Aufdeckung(abhängig vom wert der Information)
  - Informationsdiebstahl oder Zerstörung als Ziel
- Regierungsorganisationen
  - Industriespionage zum Wohl einheimischer Firmen
  - Militärspionage udn hybride Kriegsführung

1.2 Angriffe gegen einen Computer

Informationsdiebstahl führt zu:

- Wettbewerbsvorteilen
- Verwirrung
- Erpressung

Zerstörung führt zu:

- Spaß und Selbstverherrlichung
- Politischen Stellungnahmen

Sammlung von Informationen

- Infos werden zu Angreifer gesendet
- an Netzwerk angeschlossene Rechner mit höherem Risiko
- Zugriff für Angreifer durch:
  - Social engineering
  - Viren/Trojaner/Würmer
  - Physischer Diebstahl von Datenträgern
  - Sniffing

Zerstörung von Infos

- Infos gehen verloren
- physische Angriffe/Feuer/Naturkatastrophen
- Beabsichtigte Löschungen durch
  - Social Engineering
  - Viren/Trojaner/Würmer

Viren

- Infektion von Dateien

- Infektion von System und Boot record
- Zerstörung, Verwirrung und öffentliche Aufmerksamkeit als Ziel

Würmer

- Mailing Worms - Verbreitung durch E-Mails
- Viren/Trojaner evtl als „Nutzlast“
- Network worms - Verbreitung durch Ausnutzung von Softwaremängeln(bspw Bufferoverflows)
- Ablauf:
  - Zielauswahl
  - ausnutzen(exploit)
  - Infektion
  - Verbreitung

Backdoors und Trojaner

- Schadsoftware wird in nützlicher Software versteckt
- mögliche Funktionen:
  - mitschneiden von Daten(logging)
  - Zerstörung
  - Installation weiterer Software(DoS Clients, root kits etc)
  - bedingter Start von Prozessen (time bombs)

Identitäts Spoofing

- Angreifer übernimmt die Identität von jemand anderem
- Angreifer und Ziel müssne normalerweise ein Netzsegment teilen
- Angreifer liefert evtl falsche Infos über Routen oder Namen
- Grundsätzlich sind alle Antworten eines Protokolls potentielle Spoofingsubjekte(subject of spoofing?)

DoS

- Angreifer möchte einen Dienst der von einem Rechner oder Gerät angeboten wird überladen
- Angriffe gegen Konkurrenten, als pol/gesellschaftliche Aussage oder um andere Aktivitäten zu verbergen
- bösartige Anfragen sind nicht von normalen Anfragen zu unterscheiden
- BSP: HTTP, DNS DoS, SYN Flooding

Bot Network

- Fernsteuerung mehrerer Rechner um bösartige Aktionen auszuführen
- bsp: DDoS, aufwändige Entschlüsselungen berechnen

Password/Schlüssel Attacken

- Brute Force
- Raten/ Wörterbuchangriffe
- Mängel in der Implementation(z.B. Password als Klartext gespeichert)

Port/Network Scanning

- während der Aufklärungsphase um Sicherheitslücken und geeignete Ziele zu finden
- Angreifer möchte Informationen über das System erlangen
- Sniffing/Mapping/Port Scans

Session Highjacking

- Angreifer bricht in einen bestehender Session ein ohne sich einloggen zu müssen.

ZSF: viele unterschiedliche Angriffsmöglichkeiten $\Rightarrow$ unüberschaubare Anzahl an verschiedenen Attacken Angreifer unterscheiden sich in Motivation und Möglichkeiten.

2 Einführung und Rekapitulation

2.1 Sicherheitskomponenten

- Firewalls
- Intrusion Detection/Prevention Systems
- Proxies
- interne oder private Netzwerke / Netzwerkzonen / Entmilitarisierte Zonen
- VPNs

2.2 Firewalls

entscheidet ob Verkehr ins Netzwerk gelangen darf oder nicht Typen:

- Paketfilter
- Zustandsbehaftete Firewall
- Proxyfirewall

2.3 Intrusion Detection Systems

Identifizierung von Attacken / verdächtigem Verkehr
Hilfe beim Einrichten/ konfigurieren von Firewalls
Normalerweise transparent für Nutzer und Angreifer.
hauptsächlich 2 Arten:

- Mustererkennung
- Anomalieerkennung

2.4 Proxies

strickte Trennung von internen und externen Netz
Üblicherweise auf Application Layer. Verhindert dass bestimmte Informationen(Viren, Pornos, illegale Infos) in das interne Netz gesandt werden.
Verhindert, dass bestimmte Informationen nach außen gesendet werden.
Kombinationen mit anderen Systemen(Virenfilter/ Spamfilter / IDS...)

2.5 VPN

VPNs erschaffen einen gemeinsamen Addressraum.
VPNs schützen die Kommunikation über ungesicherte Netzwerke als würde sie in einem Netzwerk stattfinden.
Gegenseitige Authentifizierung der Kommunikationspartner.
VPNs bieten signifikante Einsparungen über dedizierte Verbindungen.

2.6 Zonen - DMZ

kleine Netzwerke, welche öffentlich erreichbare Dienste beeinhalten (z.B. HTTP)
DMZ oft durch Firewalls etc geschützt.
DMZ befinden sich außerhalb des internen Netzes.
sind unsicherer als das interne Netz.
Abgeschirmte Teilnetze sind isolierte Netze innerhalb des internen Netzes

2.7 internes Netz

eingeschränkter Zugriff auf das externe Netz nur über gut bekannte Ports
Internes Angriffsrisiko hängt ab von:

- Anzahl der Nutzer
- Vertrauen in die Nutzer

- Zugriffswege der Nutzer(Notebooks?)
- Fähigkeiten der Nutzer

Hosts müssen trotzdem noch mit firewalls etc geschützt werden

2.8 Basis Kryptografie

Kerkhoff's Prinzip: Sicherheit hängt nur von Schlüssel ab und nicht von der Kenntnis der kryptografischen Funktion.

2.8.1 symmetrische Verschlüsselung

Beide Teilnehmer benutzen zum ver- und entschlüsseln denselben Schlüssel.
Stromchiffren: Klartext wird Zeichen für Zeichen ver- und entschlüsselt.
Blockchiffren: arbeitet mit festen Blockgrößen und entschlüsselt mehrere Zeichen in einem Schritt.

One-Time Pads Stromchiffre deren Schlüsselstrom ein Strom aus echten Zufallsbits ist
Uneingeschränkt sicher(einziges bisher „bewiesen" sicheres Verfahren).
Schlüssel muss zu verschlüsseln mindestens so lang sein wie der Klartext.
Jeder Schlüssel darf nur einmal verwendet werden.
Nachteil: viel Speicherbedarf für Schlüssel.

2.8.2 asymmetrische Verschlüsselung

Sender und Empfänger nutzen jeweils unterschiedliche Schlüssel.
Es ist schwierig den Entschlüsselungsschlüssel(k') aus dem Verschlüsselungsschlüsselungsschlüssel(k) zu berechnen
k kann öffentlich gemacht werden (public-key-Verschlüsselung).
Nachteil: Verteilung der Schlüssel

2.8.3 hybride Verschlüsselung

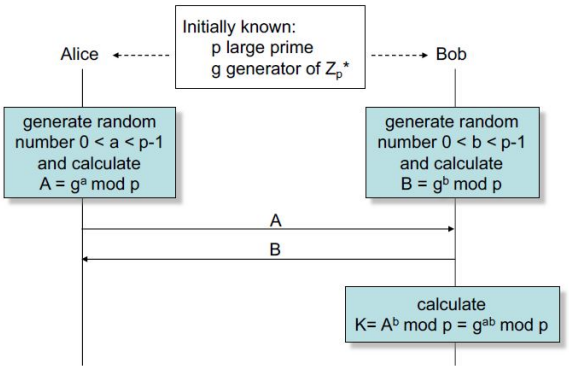
Kombination aus symmetrischer udn asymmetrischer Verschlüsselung.
symmetrischer Session Key mit dem die Daten symmetrisch verschlüsselt werden.
Session Key wird asymmetrisch mit public Key des Empfängers verschlüsselt.
löst Verteilungsproblem der asymmetrischen und behält Geschwidigkeit der symmetrischen Verschlüsselung

2.8.4 kryptografische Hashfunktion

Anforderungen:

- einseitig: wenn Hashwert y gegeben ist, ist es rechnerisch unmöglich eine Nachricht x zu finden, sodass $h(x) = y$
- schwacher Kollisionswiderstand: bei gegebener Nachricht ist es rechnerisch unmöglich eine andere Nachricht mit gleichem Hashwert zu finden
- starker Kollisionwiderstand: Es is rechnerisch unmöglich zwei Nachrichten mit gleichem Hashwert zu finden.

Hash und Signaturen Von Nachricht wird Hash gebildet. Dieser wird verschlüsselt und als Signatur an die Nachricht gehängt.
Empfänger entschlüsselt Signatur mit public key des Senders und vergleicht mit dem hash der Nachricht. Wenn gleich dann ist alles gut, wenn nicht dann wurde was verändert.



2.8.5 Diffie-Hellman Schlüsselaustausch

2.8.6 Zertifikate

Zertifikat ist eine Datenstruktur welche folgendes enthält:

- Öffentlichen Schlüssel
- Namen des Eigentümers des öff Schlüssels
- Namen des Ausstellers
- Ausstellungsdatum
- Ablaufdatum
- Möglicherweise andere Daten
- Signatur des Ausstellers

2.8.7 Certification Authorities (CA)

stellen Zertifikate aus.
sind normalerweise vertrauenswürdige Dritte.
Zertifikate werden über online Datenbanken verteilt(Certificate Directories) denen vertraut werden muss.

3 Paketfilter

3.1 Funktionsweise von Paketfiltern

Netzwerkpakete werden akzeptiert oder zurückgewiesen anhand von Parametern wie:

- Quelladresse/Ports
- Zieladresse/Ports
- Flags

3.2 Paketfilterregeln

Regeln können bzgl Flags, Adressen und Ports angewandt werden.
Paketfilter können auch bezüglich des Inhaltes von Paketen angewandt werden.
Zulassende Regeln:

- explizites Erlauben von Zugriff
- sämtlicher anderer Verkehr wird verhindert

Verhindernde/ablehnende Regeln:

- bestimmter Verkehr wird explizit abgelehnt.
- sämtlicher anderer Verkehr wird für gewöhnlich zugelassen.

Wichtig:

- Reihenfolge der Regeln ist wichtig.(erst alles verhindern und dann einige zulassen ist was anderes als erst einige zulassen und dann alles zu verhindern!)
- große Anzahl an Regeln kann verwirrend sein.
- „alles verbieten und solange es nicht explizit benötigt wird“ kann gute Herangehensweise sein.

3.2.1 Ingress-Filter

Filtern ankommende Pakete
blockieren Zugriff von verdächtigen Quelladdressen.

3.2.2 Egress-Filter

Filtern ausgehenden Verkehr.
Nur Pakete mit Quelladdresse im Netzwerk dürfen das Netzwerk verlassen so lange keine andere Regel greift.
Quellen von abgewiesenen Paketen sind gute Kandidaten für Überprüfung.

3.2.3 Protokollfilter

Dienste haben haben festgelegte Protokolle
Daumenregel: nur Verkehr zu Diensten zulassen die wirklich benötigt werden.

3.2.4 Probleme

Zugriff für bestimmte Netze zulassen
Gefahr des Spoofings: Angreifer nutzt evtl falsche Quelladressen
Source Routing:

- Pakete enthalten evtl Infos über die Route zurück zum Urheber
- Überschreiben die Routingtabelle des Routers

Gefahr das Filterregeln umgangen werden
Fragmentierung:

- Paketfilter untersuchen Headerinfos
- Paket wird so aufgeteilt dass der Header geteilt wird und Adresse und Ports nicht gefiltert werden können.

Löcher:

- Dienste müssen erreichbar bleiben für externe Netzwerke
- entsprechende Ports müssen geöffnet werden.

3.3 dynamische Paketfilter

Filterregeln werden on the fly so erstellt wie sie benötigt werden und nach schließen der Verbindung wieder gelöscht.
Filter beobachten ausgehenden Verkehr und erstellen zurückwirkende Regeln.(Ausgehender Verkehr zu einer Adresse bewirkt Regel dass eingehender Verkehr von dieser Adresse erlaubt wird.)
Probleme:

- Regeln sind angreifbar z.B. durch senden falscher reset Pakete
- ausgehender Verkehr wird nicht gefiltert. Gefahr von Trojanern/Viren.

4 Zustandsbehaftete Firewalls

4.1 Funktionsweise

Kennen den Zustand von Verbindungen und wissen welche Pakete in welchem Zustand erwartet werden.
Es können Regeln angewandt werden die nur in bestimmten Zuständen wirksam sind.
Untersuchen hauptsächlich OSI 4 (transport layer), aber auch höhere Schichten.

4.2 Probleme

Hohe Leistung benötigt teilweise geclusterte Hardware. Zustandsbehaftete Firewalls lassen sich nicht einfach clustern.
Zustandslose Protokolle (UDP, ICMP, DNS, HTTP)

4.2.1 Zustandslose Protokolle

Zustandslose Protokolle definieren trotdem welche Pakete erwartet werden.
Timeouts werden genutzt um Pseudo-Verbindungen zu erzeugen.

4.3 Multi-Layer Inspection

Die meisten Protokolle basieren auf Protokollen aus niedrigeren Layern. BSP: HTTP nutzt TCP Verbindungen. Zustandsbehaftete Firewalls können beide Layer beobachten.

5 Proxy Firewalls

Proxies verhalten sich für den inneren Client wie der äußere Server und andersherum. Client und Server werden niemals direkt miteinander agieren. Der Proxy ist außerdem intransparent. Auf dem Proxy selbst können ein paar Programme laufen, die sicher sind und denen vertraut werden kann.
Proxies können die interne Struktur eines Netzwerkes nach außen hin verstecken und gegen Protokoll-Angriffe schützen. Forward Proxies sind ein Mittel, um ausgehenden Traffic zu kontrollieren - Reverse Proxies für eingehende Verbindungen.

Beispiel: Der Nutzer fragt eine HTTP-Ressource an. Dessen Software leitet die Anfrage an den Proxy weiter. Der Proxy baut die Verbindung auf und gibt sich als Client aus, der die HTTP-Ressource beim Server anfragt. Sämtlicher Traffic zwischen dem internen Nutzer und dem externen System wird durch den Proxy geleitet.

5.1 Bastion Host

Unter einem **Bastion Host** versteht man in diesem Kontext einen Proxy der auf das öffentliche Internet zugreift und daher besonders gegen Angriffe geschützt und abgehärtet werden muss. Dies ist notwendig, da der Proxy-Server von außerhalb des Netzwerks sichtbar ist. Da Proxies kein IP-Forwarding machen, besitzen sie üblicherweise mit zwei Network-Interfaces. Die interne Struktur des Netzwerks bleibt vor äußeren Einblicken gesichert. (Passive Fingerabdrücke sind nicht möglich)

5.2 Arten von Proxies

- **Forward Proxy**: Weitverbreitetste Form, bei der die Verbindung vom internen Client initiiert wird
- **Reverse Proxy**: Hier wird die Verbindung von der externen Seite initiiert. Wird von Sicherheitsservices genutzt: Der ankommende Datenverkehr wird vom Proxy überwacht.
- **Application-Level Proxy**: Bietet für jeden Service Policies, die bestimmten Traffic genehmigen (bspw. Nutzer, Adressen, Computer,...)
- **Transparent Proxy**: Ein transparenter Proxy (intercepting proxy, inline proxy, or forced proxy) leitet die normale Kommunikation für beide Seiten normal auf dem Network-Layer weiter. Es ist keine spezielle Konfiguration notwendig. Der Client muss sich der Existenz des Proxies nicht bewusst sein.
- **Intercepting Proxies** Diese „abfangenden“ Proxies werden üblicherweise benutzt um Policies durchzusetzen, ohne dass eine clientseitige Browserkonfiguration notwendig wäre. Bspw. Nacktfilter
- **Intransparente Proxies** Für diese Art muss der Nutzer sein Gerät anpassen, um den Proxy nutzen zu können.
- **Circuit-level Proxy** Die Filterung dieses Proxies wird durch speziellere Regeln definiert. Der Inhalt wird auf Schlagwörter, Größe, Viren, Datentypen, Bilderkennnung, Passwörter oder ähnliches geprüft. Des Weiteren ist Authentifikation/ Legitimation möglich, um bestimmten Nutzern Rechte einzuräumen.

5.3 SOCKS

Bei SOCKS handelt es sich um ein Proxy-Toolkit. Es ermöglicht, dass Anwendungen sich ohne spezielle Client-Software mit Proxies verbinden können. Ein SOCKS-Server führt Client Authentifizierung und Authorisierung durch. Zur Kommunikation mit einem SOCKS-Server sind jedoch Modifikationen notwendig.

Der Client sendet einen Request an den Server. In diesem Request sind die Identität des Clients, die Ziel-Adresse (Ausgehende Verbindungen) ODER der Port (Eingehende Verbindungen.
Der Proxy prüft (als Server), ob der Request bewilligt werden soll. Im positiven Fall, antwortet er mit einem Reply-Paket, welches den Return-Code der Operation beinhaltet.
Protokoll:

- Der Client möchte sich mit einem externen Service verbinden und sendet diesen Request:
  | VN | CD | DSTPORT | DSTIP | USERID | ... | NULL |
  (VN-Versionsnummer, CD-Command)
- Reply | VN | CD | DSTPORT | DSTIP |
  (90: request granted, 91: request rejected or failed, 92: request rejected becasue SOCKS server cannot connect to identd on the client, 93: request rejected because the client program and identd report different user-ids)
- Client bietet eine eingehende Verbindung an und sendet einen Bind-Request
  | VN | CD | DSTPORT | DSTIP | USERID | ... | NULL |

5.4 Umgang mit Verschlüsselung

Wenn die übertragenen Daten verschlüsselt sind, ist eine Ende-zu-Ende-Verschlüsselung nicht möglich. Der Proxy kann die übertragenen Daten nicht verändern, wenn diese verschlüsselt sind. Deshalb ist es unmöglich den Inhalt zu überprüfen.
Es kann zu Problemen bei der Authentifizierung kommen, wenn zwischen Server und Client ein Proxy sich befindet. Diese Probleme treten insbesondere dann auf, wenn dieser versucht die Daten zu entschlüsseln.
Der Proxy kann jedoch als Client agieren. In diesem Fall überträgt er die Daten entschlüsselt zum Client und verschlüsselt zum Server.

5.5 Diskussion

Vorteile:

- Schutz der internen Struktur nach Außen
- Datentraffic kann einfach überwacht werden
- Nutzerbezogene Sicherheit ist möglich
- Authentifikation kann implementiert werden
- Schützt vor Spoofing (Der Proxy generiert sämtlichen ausgehenden Traffic für einen Außenstehenden)

Nachteile:

- Leistungseinbußen
- Single Point of Failure/Attack
- Anwendungsspezifische Proxies müssen für jede einzelnen Anwendungen entwickelt werden
- Software muss angepasst werden
- Bastion Host muss gehärtet werden

6 Policies

6.1 Was sind Policies?

Eine Sicherheitsrichtlinie(security policy) beschreibt was getan werden muss um auf einem Rechner gespeicherte Infos zu schützen.
Richtline definiert was gemacht werden muss und und wie es ausgewertet werden kann.
Werden normalerweise aufgeschrieben.

6.2 Komplexität von Richtlinien

werden von Menschen definiert.
müssen verständlich formuliert sein.
zu komplexe (aber auch zu einfache. bsp „keine rechner benutzen!“) Richtlinien sind nicht durchsetzbar.

6.3 Entwicklung von Richtlinien

beste Vorgehensweise:

- Risiken identifizieren
  - Sicherheitsanalyse
    - kritische Daten und Systeme identifizieren
    - normale Nutzung des Netzwerkes feststellen
  - aufschreiben
- Funde kommunizieren
  - Dem Management berichten
    - einfach
    - ausgeglichen
    - präzise
    - Zeigen auf einzelne vermeiden
    - Allgemein halten
- Richtlinie erstellen oder aktualisieren
  - aufschreiben
  - Genauheit und Klarheit
    - Was muss getan werden?
    - Warum?
    - Wer ist verantwortlich?
  - Knappheit: Niemand liest mehr als 10 Seiten.
  - Realismus
- Einhaltung der Richtline kontrollieren
  - Wenn die Einhaltung einer Regel nicht kontrolliert werden kann ist sie nicht durchsetzbar
  - Stichproben, Log Analysen, Festplattendurchsuchungen
- Versuchen eine „Kultur“ zur Einhaltung der Richtlinie einzuführen
  - Anpassung an die Richtlinie hängt stark vom Verhalten der Nutzer ab
  - Mit Nutzern über Risiken sprechen
  - Richtlinien vor der Einführung erklären
  - Anweisungen und autoritäres Verhalten vermeiden

6.4 ungeschriebene Richtlinien

versteckte Regeln existieren.
nutzen des gesunden Menschenverstandes
Versuchen ein Sicherheitsbewusstsein im Betrieb zu etablieren.
Wissen verbreiten, aber vorsichtig und sensibel.

7 Intrusion Detection Systems (IDS)

Sind dazu bestimmt Angriffe zu erkennen und nicht um diese zu verhindern.
Netzwerk IDS Sensor liest Traffic mit uns analysiert diesen indem nach Zeichen für:

- Scans/Sonden
- Aufklärungsaktivitäten
- Exploits

Ist normalerweise komplett transparent (nicht zu entdecken).

7.1 Motivation

Ohne IDS würde ein Admin die meisten Angriffe nicht bemerken und kann auf diese somit nicht reagieren.
Fehlende Infos ohne IDS:

- Welche Hosts wurden angegriffen?
- Welche Daten wurden kompromittiert?
- Mit welcher Methode wurde angegriffen?

Nachfolgende Schritte eines Angriffes können verhindert werden.

7.2 Methoden

7.2.1 Anomalieerkennung

- Statistische Analyse um „unnormalen“ Verkehr erkennen zu können.
- Parameter wie Herkunft, Datenrate, Ports und Zeit werden berücksichtigt und gegen eine Statistik geprüft, jedoch nicht nach bestimmten Mustern.
- Berechnung der Wahrscheinlichkeit dafür das Verkehr unnormal ist mithilfe von Bayesschen Filtern.
- Training der Filter mit Verkehr der als normal betrachtet wird.
- Wenn eine bestimmte Grenze überschritten wird, wird Alarm ausgelöst.

7.2.2 Signaturerkennung

- Analyse von Paketen anhand von gegebenen Mustern.
- Adressen und timing Pattern werden berücksichtigt.
- bestimmte Mustern lösen einen Alarm aus.

7.3 Probleme mit IDS

Fehlalarme (false positives) und nicht erkannte Alarme (false negatives).

- Viele Fehlalarme werden zu einer bedeutend höheren Ignoranz seitens des Admins führen.
- Reduzierung der false positives führt gewöhnlich zu mehr false negatives.
- IDS Evasion um false positives zu verringern
- Als ersten Filter ein allgemeines Muster Nutzer
- Spezifischere Untersuchung der Pakete die den ersten Filter durchlaufen haben.

7.4 Ansätze für IDS

Die Auflösung der Referenzmenge wird betrachtet, um ein bestimmtes Datenobjekt als Außenseiter zu erkennen.

- globale Ansätze:
  - Referenzmenge enthällt alle Daten.
  - Basisannahme: es gibt nur einen normalen Mechanismus (normal mechanism?).
  - Problem: andere Ausreißer sind auch in der Menge und können das Ergebnis verfälschen.
- lokale Ansätze:
  - Die Referenz enthält lediglich eine Teilmenge der Daten.
  - keine Annahme über die Anzahl von normalen Mechanismen (normal mechanism?).
  - Problem: Wie fählt man eine geeignete Referenzmenge

Einige Ansätze befinden sich irgendwo dazwischen.
Die Auflösung der Referenzmenge kann automatisch oder durch eine Nutzereingabe verändert werden.

7.5 Verfahren

7.5.1 Überblick

- Vergleich Labeling vs. Scoring
  - Beim Labeling werden Objekte entweder als normal oder outlier eingeteilt.
  - Beim Scoring wird jedem Objekt ein Wert zugeordnet, der die Wahrscheinlichkeit angibt, ob es sich bei dem Objekt um einen Outlier handelt).
- Modellbasiert
  - Rational: Normale Datenpunkte werden durch dieses Modell repräsentiert - Outliers sind Punkte die davon abweichen
  - Ansatz der auf Proben basiert: Wahrscheinlichkeitstests, die auf statistischen Modellen basieren.
- Nähebasiert
- Winkelbasiert

7.5.2 Statistische Tests

Idee:

- Gegebene Wahrscheinlichkeitsverteilung (z.B. Gauss)
- Parameter berechnen unter der Annahme, dass alle Datenpunkte durch eine solche Wahrscheinlichkeitsverteilung generiert wurden.
- Ausreißer sind die Punkte, die eine geringe Wahrscheinlichkeit haben von der Verteilung generiert worden zu sein (z.B. weicht mehr als 3 mal von der Standardabweichung ab).

Annahme:

- Normale Daten folgen einer Verteilung und treten in einer Region des Models mit hoher Wahrscheinlichkeit auf.
- Ausreißer weichen stark von dieser Verteilung ab.

Viele verschiedene Tests (unterschiedliche Verteilung, Menge der Variablen, Menge der Verteilungen....)

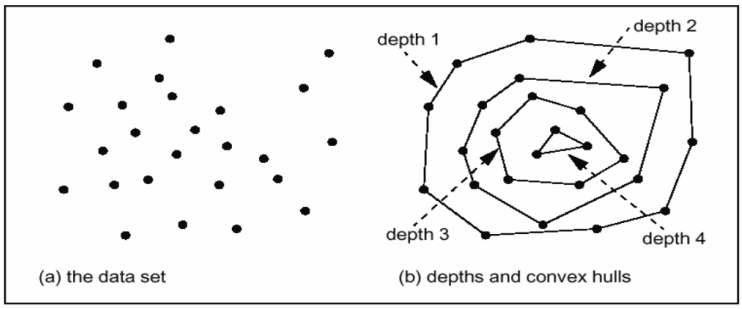
7.5.3 Tiefenbasierte Ansätze

Idee:

- Suche nach Ausreißern an den Grenzen des Datenraums aber unabhängig von Wahrscheinlichkeitsverteilungen.
- organisieren Daten in Schichten von konvexen Hüllen
- Ausreißer sind Objekte in äußeren Schichten.

Annahme:

- Ausreißer befinden sich an den Grenzen des Datenraums und normale Daten im Zentrum.



7.5.4 Abweichungsbasierte Ansätze

Idee:

- Gegeben: Menge von Datenpunkten
- Ausreißer sind Punkte die nicht in die allgemeinen Charakteristiken der Menge passen.

Annahme:

- Ausreißer sind die äußersten Punkte der Datenmenge.

7.5.5 Abstandsbasierte Ansätze

Idee:

- Punkte werden basierend auf ihrem Abstand zu ihren Nachbarn bewertet.

Annahme:

- normale Daten haben eine dichte Nachbarschaft
- Ausreißer sind weit von ihren Nachbarn entfernt.

Beispiel: k-Nearest-Neighbours

- Man nehme die Summe der k nächste Nachbarn und nutze diese als Outlier-Score
- Die Knoten mit einem geringen Score sind besser angebunden als die mit einem höheren
- Höherer Score → Indiz für Outlier

7.5.6 Dichtebasierte Ansätze

Idee:

- Vergleiche Dichte um einen Punkt mit der Dichte um seine lokalen Nachbarn
- Die relative Dichte eines Knotens im Vergleich zu der seines Nachbarn wird als Ausreißerwert berechnet.
- Ansätze unterscheiden sich in der Bewertung der Dichte

Annahme:

- Die Dichte um normale Daten ist gleich der Dichte um seine Nachbarn.
- Die Dichte um einen Ausreißer ist erheblich anders als die Dichte um seine Nachbarn.

8 Honeypots and Tarpits

8.1 Honeypot

Antivirus Software Hersteller möchten neue Viren und Varianten kennenlernen

- Forschungshoneypots
- gewöhnlich große verteilte Netzwerke von Fallen

Netzwerkadmin möchte Informationen über aktuelle Bedrohungen erlangen

- gewinnbringende (productive?) Honeypots
- um Infos über bekannte Angriffstypen zu erlangen

8.1.1 Honeypots um Angreifer abzulenken

Admin möchte Angreifer vor verletzlicheren Zielen ablenken Aus Netzwerksicherheitstechnischer Sicht keine akzeptable Methode!

8.1.2 Arbeitsprinzipien

Honeypots verhalten sich wie reale Systeme.
Angreifer sollen nicht mitbekommen können, dass sie mit einem Honeypot interagieren.
Honeypot zeichnet alle Nutzer/Client Handlungen auf.
Admin möchte den Spuren eines Angreifers folgen.
Später folgt eine Analyse der Angriffsmuster.

8.2 Honeynets

Reale Systeme hinter einem Gateway beobachten alle Netzwerkaktivitäten.
Es können nur Netzwerkaktivitäten beobachtet werden, lokale Aktivitäten (Viren etc.) sind nicht interessant.

8.3 Spam Traps

E-Mail Adressen werden in gut sichtbaren Bereichen platziert.
Alle Nachrichten die an diese Adressen geschickt werden, werden als Spam betrachtet, da die Köder nicht für echt Anwendungen benutzt werden.
Später Blacklisting der entsprechenden Adressen und Sender.

8.4 Virus Traps

Antivirus Forscher zeichnen alle Virusaktivitäten in einer „Sand Box“ auf.
Große Vielfalt an Systemkonfigurationen ist notwendig.
Gegenmaßnahmen müssen sehr schnell (normalerweise innerhalb von 6 Stunden) entwickelt werden.

8.5 Tarpits

Wie Honeypots, jedoch mit aktiver Funktion.
Verlangsamen Angreifer indem Warteperioden in Protokolle eingeführt werden.
Reduzieren die Verbreitungsgeschwindigkeit von Würmen.
Erhöhen die Kosten für die Angreifer.
BSP: Tarpit verzögert die SMTP Antworten solange, dass fast der Timeout eintritt.

8.6 Proof-of-work Systeme

Proof-of-work (POW) fügt einem Dienst einen Preis hinzu.
ökonomische Maßnahme um DoS Angriffe oder Spam zu verhindern.
benötigen für gewöhnlich Rechenzeit.
Können als Nebenprodukt für praktische Rechenaufgaben benutzt werden (Jeder Sender einer Mail rechnet also ein kleines bisschen an einem Problem das eh gelöst werden soll weiter).

9 VPN

- Funkiton von VPN: Verbinden mindestens zweier Endpunkte bzw. mehrerer Netzwerke und zum Erstellen eines gemeinsamen Adressraumes.
- Zwei Arten:
  1. Host-to-Gateway: Einbindung von Remote-Nutzern
  2. Gateway-to-Gateway: Aufbau eines verschlüsselten Tunnels zwischen den Gateways

9.1 Host-to-Gateway Varianten

- Voluntary tunneling
  - Der Nutzer kann entscheiden, ob er via VPN auf eine Netzwerkressource zugreifen will oder nicht.
  - Vorteil: Bedeutet weniger Aufwand für das VPN-Gateway
  - Nachteil: Der Nutzer ist zeitgleich mit dem inneren und äußeren Netzwerk verbunden. Dadurch ist der Nutzer nicht durch die Sicherheitsmittel des inneren Netzwerks gesichert.
- Compulsory tunneling
  - Der gesamte Traffic wird durch das VPN-Gateway geforwarded.
  - Der Nutzer kann keine alternative Route nutzen.
  - Vorteil: Der Nutzer durch die Sicherheit um das innere Netzwerk gesichert ist
  - Nachteil: Das VPN-Gateway muss mehr Traffic verarbeiten

9.2 Protokolle

- Tunneling-Protokolle kapseln Protokolle ineinander. Das äußere arbeitet als Data-Link-Layer für das innere Protokoll.
- Beispiele: L2TP, PPPoE, IPSec, 802.1Q, SSL / TLS

9.3 Angriffe gegen:

- Verschlüsselte Pakete und Authentifizierung
  (Erfordert Kryptoanalyse, Expertenwissen und die meisten Verschlüsselungsverfahren sind genau untersucht und können grundsätzlich als sicher angesehen werden)
- Client
  (Angreifer wollen Zugriff erhalten, bevor die Daten verschlüsselt werden oder nachdem die Daten entschlüsselt wurden - Trojaner)
- Gateway
  (Spoofing)

9.4 Pro/Cons

Vorteile

- Sicherheit (Authentifizierung, Verschlüsselung)
- Entwicklungsgeschwindigkeit (Es ist nicht nötig auf eine neue Leitung zu warten)
- Kosten (Nutzung(Teilen) eines existierenden Netzwerks)

Nachteile

- Overhead (Weiterer Arbeitsschritt, Tunneln, Entkapseln)
- Implementation (VPN muss in eine existierende Struktur integriert werden)
- Verschlüsselte Dateien können nicht gefiltert werden

9.5 Design Entscheidungen

- Jedes Paket muss authentifiziert sein, um Session Hijacking zu verhindern
- Daten müssen vor der Verschlüsselung auf Integrität hin untersucht werden
- VPN bewirkt keine Firewall-Funktionalität
- Eine Firewall kann verschlüsselten Traffic nicht untersuchen

10 Public Key Infrastructure

Ziele/Zwecke von PKI:

- Erzeugung,
- Verteilung und
- Widerrufung von Zertifikaten.

- Unterstützung bei sicherer Kommunikation und in der Handhabung von rechtlich bindenden Dokumenten (Signaturen, Nichtabstreitbarkeit).
- Komponente im DRM System

Public Key Kryptographie: Ein Schlüssen zum Verschlüsseln, ein anderer zum Entschlüsseln.
bekannte Algos sind: Diffie-Hellman(DH) und RSA.

10.1 Zertifikaterstellung

Nutzer erstellt Schlüsselpaar(Als Datei oder auf einer Smartcard).
Public Key muss authentifiziert werden.

- CA signiert den Public Key und generiert damit ein Zertifikat.

10.2 PKI Protokolle

X.509:

- ermöglicht Interoperabilität zwischen mehreren Anwendungen (Webserver, Mail tool, VPN Gateway).

LDAP:

- wird herkömmlicherweise genutzt um X.509 Zertifikate und Revocationlists der PKI zu verteilen.

10.3 Zertifikatwiderruf

Zertifikate können öffentlich verteilt hochverfügbar gemacht werden.
Private Keys können verloren gehen oder gestohlen werden.

- verlorene Keys können erneut generiert werden.
- gestohlene Keys stellen eine Sicherheitsbedrohung dar.

CA unterhält eine Liste der widerrufenen Keys- Certificate Revocation List (CRL)

10.4 Zertifikatsverwaltung

normalerweise stark reguliert (durch Firmenrichtlinien oder nationale Gesetze).
Funktionen:

- öffentlicher Aufbewahrungsort für Zertifikate (LDAP).
- Ablage für Revocation List.

10.5 Schlüsselverwaltung

private Keys müssen gegen Diebstahl und Verlust geschützt werden.
Manche PKI Systeme generieren Keys für Nutzer und liefern diese zu ihnen.
Chronik der public Keys muss gepflegt werden.

10.6 Client Software

interagiert mit Server Komponenten.
erlaubt Schlüssel- und Zertifikatsverwaltung.

10.7 Hardware Tokens

private Keys sind gefährdet wenn die auf Festplatten gespeichert sind.
private Keys können nicht einfach verteilt werden.
Tokens schützen Schlüssel und bieten zusätzliche Sicherheit, da der Nutzer ein physikalisches Token vorliegen hat.
Formen: Smartcards, RFID Tags, USB Token, iButton.

11 Enterprise Authentication

Authentifikation:

- Prozess in dem versucht wird die Identität oder den Anspruch (claim?) von jemanden zu verifizieren.

Bedarf an Authentifikation:

- Login Prozess / Zugriffskontrolle
- Sicherheitsprotokolle, Web of Trust

11.1 Authentifikationsprinzipien

Wissensbasiert (PIN, Password, Keyphrase, persönliche Infos)

- geteilte geheime Information muss dem Authentikator präsentiert werden.
- Probleme:
  - Nutzer wählt schwaches Passwort
  - Nutzer vergisst Passwort
  - Password ist kompromittiert.

Besitzbasiert (SmartCard, SIM Card, Kreditkarte, Handy, PC etc)

- Die Tatsache, dass jemand im Besitz von etwas bestimmten ist, muss über das Netzwerk verifiziert werden.
  - Es muss eine geheime Information im Token geben.
  - Geheime Info an sich sollte nicht über das Netzwerk transportiert werden.
  - Nutzer können das Token verlieren.
  - Token kann kopiert werden.

Biometrisch (Retina, Iris, Fingerabdruck, DNA, Stimme...)

- Biometrische Eigenschaften können nicht vergessen, gestohlen, abgehört oder gelernt werden.
- Handhabung ist z.T. kompliziert.
- Normalerweise die einzige Methode um eine Person an ein Gerät zu binden. Alle anderen Methoden können auch irgendwie von anderen Personen durchgeführt werden.

Erhöhte Sicherheit durch 2-Faktor.Authentifizierung.

- 2 der oberen Methoden kombiniert
- aka „strong authentication“

11.2 Probleme

Wie authentifiziert man sich gegenüber jemanden der sich selber nicht authentifiziert hat?

- Könnte Man in the middle sein.
- Replay Attacken müssen verhindert werden, benötigen gegenseitige Authentifizierung.

Wie authentifiziert man sich über Wissen oder Besitz ohne das Geheimniss / den Beweis offen zu übermitteln?
Standardisierte Protokolle für viele Anwedungen.

11.3 Angriffe

Man in the middle:

- Jemand „sitzt“ zwischen 2 Kommunikationspartnern und spielt beide Rollen.
- Es wird ein vertrauenswürdiger Dritter benötigt um diese Angriffe zu verhindern.

Replayattacken:

- Jemand zeichnet eine verschlüsselte Authentifizierungsnachricht auf und sendet diese ebenfalls um Zugriff zu erlangen.
- Ein einzigartiger Wert (im optimalen Fall nur einmalig genutzt) wird in jeder Nachricht benötigt.

Denial of Service:

- Vandalismus
- jemand versucht die Kommunikation zu verstopfen.
- Für gewöhnlich schwer zu besiegen/verhindern.

11.4 Klartextpasswörter

Einige Protokolle nutzen unverschlüsselte Passwörter (Telnet, FTP, POP3, IMAP4).

11.5 Challenge-Response

Geteiltes Geheimnis an sich soll nicht übermittelt werden, aber das Wissen soll bewiesen werden.
Replay Attacken sollen verhindert werden. Ablauf:

- Antragsteller fragt beim Server um Zugriff an.
- Server schickt eine Challenge.
- Antragssteller berechnet die Antwort (normalerweise ein kryptografischer Hashwert).
- A sendet die Antwort.
- Server vergleicht Antwort mit Ergebnis und gewährt Zugriff.

11.6 Authentifikationsmethoden

11.6.1 Kerberos

Computer Netzwerk Authentifizierungsprotokoll
erlaubt es Einzelnen über ein unsicheres Netzwerk zu kommunizieren um ihre Identität gegenüber jemand anderen auf einem sicheren Weg zu beweisen.
Die Entwickler richteten sich hauptsächlich an Client-Server-Modelle und unterstützen gegenseitige Authentifizierung.
Nachrichten sind gegen Abhörung und Replay Attacken abgesichert.
Bestandteile:

- Client
- Authentication Server
- Ticket granting Server

- Service Server

Ablauf (stark zusammengefasst! Details stehen in Folie 110 S.19ff. Laut Thomas jedoch nicht prüfungsrelevant!):

- Authentifizierung
- Authorisierung
- Anfrage

11.7 RADIUS

wie Kerberos für Authentifizierung, Autorisierung und Abrechnung(Accounting?) der Netzwerkressourcen genutzt.
für DSL, wireless(802.1X) und VPNs verwendet.

11.7.1 CHAP

Nutzt 3-way-handshake des point-to-point Protokolls als Authentifizierung.

11.7.2 PAP

Password authentication protocol
Unverschlüsselte ASCII Passwörter
total unsicher, aber oft verwendet.
Client sendet Nutzername und Passwort.
Sever sendet authentication-ack (wenn Credentials stimmen) oder authentication-nak (sonst).

11.8 Single sign on

Methode um das Eingeben von Passwörtern für verschiedene Anwendungen zu reduzieren.

- vereinheitlichte Authentifizierung
- Speichern von Passwörtern (gefährlich)
- Speichern von Anmeldedaten (siehe Kerberos)

12 Securing Hosts and Appliances

12.1 Host Sicherheit

Jedes System das Teil eines Netzwerkes ist ist potentiellen Opfer eines Angriffes.
Übernommene Rechner innerhalb eines Netzwerkes stellen ein ernstes Risiko dar.
Falscher Glauben in die Sicherheitsmaßnahmen im Netzwerk erleichtert solche Angriffe.

12.1.1 Härten gegen lokale Angriffe

Administrative Programme begrenzen
Alle unnötigen Tools entfernen.
Für gewöhnlich ist es schwerer einen Angreifer zu stoppen nachdem er Zugriff erlangt hat.
Dateiberechtigungen:

- Richtige Datei- und Verzeichnisberechtigungen um zu verhindern, dass lokale Nutzer das System übernehmen können.
- auch genutzt um den Zugriff auf geheime Dokumente zu begrenzen.
- Andere Zugriffskontrollschemata erwägen.

Gruppenmanagement:

- Strikte Kontrolle der Mitgliederschaften der Gruppen
- Gruppen mit den niedrigst möglichen Rechten bevorzugen.

Logging:

- IDS kann Anomalien im Netzwerkverkehr nur entdecken wenn dieser das IDS passiert hat.
- Sicherheitsrelevante Infos sollten jedoch auch lokal geloggt werden.

12.1.2 Härten gegen Netzwerkangriffe

Host gegen Angriffe aus dem internen Netz schützen.
Alle bekannten Sicherheitsmaßnahmen wie Firewalls etc. bieten keinen Schutz gegen solche Angriffe.
Unnötige Accounts und Dienste entfernen:

- Effektivster Weg um erfolgreiche Angriffe zu verhindern.
- Jeder laufende Dienst kann potentiel fehlerhaft sein.
- Unnötige Accounts löschen um zu verhindern dass ein Angreifer sie nutzt.
- Deaktivierte Accounts können wieder aktiviert werden.

Passwort Richtlinien:

- Wörterbuchangriffe verhindern - neue Passwörter gegen bekannte Wörterbücher gegenchecken.
- Minimale Länge und Komplexität
- regelmäßigen Passwortwechsel erzwingen
- Wechsel zu früherem Passwort verhinden
- zu häufiges Wechseln des Passwortes verhindern

Public Key Authentifizierung:

- Public Key auf Host gespeichert - Nutzer muss private Key kennen um sich anzumelden
- SmartCards bieten weitere Sicherheit

Default Passwörter ändern!

12.1.3 Härten gegen Anwendungsangriffe

Übernahme des Rechners du Einbruch in Anwendungen die auf einem höherem Priveligierungslevel laufen.

- Buffer overflows
- Anwendungspasswörter (z.B. Datenbank)
- spezifische Accounts für Anwendungen

Patches:

- Jede verfügbare Software enthält Fehler
- einige von denen sind kritisch für die Systemsicherheit
- für gewöhnlich werden Patches von zu Zeit verfügbar, admin muss benachrichtigt werden.
- newsletter, automatische Updates (in manchen Fällen gefährlich)

Anti-Virus Software:

- kann als online Filter im Netzwerk laufen oder auf lokalen Rechnern.
- Muss aktuell gehalten werden um ausreichenden Schutz zu bieten.
- Gewöhnlich mit auto-Update Funktion
- manche agressive Viren verbreiten sich dennoch schneller.
- Anti-Virus Software kann jedoch auch selbst verwundbar sein
- Probleme mit komprimierten / verschlüsselten Dateien.

Zusätzliche Maßnahmen:

- Hashes für ausführbare Dateien berechnen um Änderungen zu bemerken
- Verdächtiges Verhalten entdecken.

12.2 Host Firewalls

Umfassende Firewalls schützen das interne Netz nur vor dem externen Netz
einige Angreifer/Malware kann sich im internen Netz befinden
Host Firewalls (personal Firewalls) arbeiten nach den gleichen Prinzipien wie die Firewalls an den Netzwerkgrenzen.

- Paketfilter
- Zustandsbehaftete Paketfilter

Host Firewalls können außerdem entdecken welche Prozess versucht Daten zu senden
Egress Filter (Filtern des ausgehenden Verkehrs) können einige Angriffe auf benachbarte Rechner verhindern.
Host Firewalls müssen entweder von einem Admin konfiguriert werden, oder sehr einfach bedienbar sein.

12.3 Host IDS

Log Generierung fütr post mortem Analyse

- Logging des Dateisystems
- Logging der Netzwerkverbindungen
- Logging der Zugriffe auf die Log Files

12.4 Sichere Verteilung von Software

Updates, Patches, Muster müssen auf alle Rechner im Netz runtergeladen werden.
Updateprozess bedeutet eine ernste Sicherheitsbedrohung (Auslieferung von Maleware auf dem Silbertablett).
Schutz der Softwareverteilung

- Digitale Signaturen

12.5 Netzwerkgeräte

Router und Switches sind zentrale Komponenten eines Netzwerkes, verantwortlich dafür den Weg der Pakete auf Data link und Network layer zu bestimmten.
Ein Router verbindet 2 oder mehrere Netzwerke
mögliche Angriffe:

- Re-routing von Paketen
  - Man-in-the-middle
  - DoS
  - Überbrückung anderer Sicherheitskomponenten
- Statistiken sammeln

12.5.1 spezielle Router

NAT Gateway - network address translation

- Router hat eine nach außen gerichtete Adresse und liefert private interne Adressen
- Interne Adressen sind nicht erreichbar bis der Router eine Adressenübersetzung (address translation) durchgeführt hat.
- Router führt diese Übersetzung nur durch wenn eine entsprechende Regel existiert.

VPN Gateway

- Router liefert Authentifizierung, Verschlüsselung und Autorisierung.
- siehe VPN

kombinierte Geräte

- Router können integrierte Firewalls, proxys etc. haben.

12.5.2 Härten

Betriebssystem das auf den Routern und Switches

- OS sollte unter speziellen Sicherheitsaspekten entwickelt worden sein
- Sicherheitsupdates sollten verfügbar sein

Zugriff zu Geräten

- Protokoll (telnet vs. ssh, http vs https)
- SNMP (Simple Network Management Protocol)
- ACL zu bestimmter Konsole
- Serieller Port für Konfiguration
- Physikalischer Zugriff
- alle nicht benötigten Dienste abschalten

Logging der sicherheitsrelevanten Vorfälle

- Zugriff zu Routerkofiguration
- Änderungen in der Konfiguration
- Ungewöhnlich hohe „Verkehrsaufkommen“

13 Organisational Aspects

13.1 Grundlagen

Wie organisiert man ein Organ um die Datensicherheit aufrechtzuerhalten?
Technische Lösungen lassen sich einfach nach und nach verstehen, müssen jedoch in professioneller und sturkturierter Art und weiße angewandt werden um Datensicherheit zu garantieren.
Komponenten eines „Informationsecurity management system“(ISMS):

- Security Process
- personnel
- management principles
- ressources

Sicherheit ist kein Zustand, es ist ein Prozess
Ein Sicherheitsprozess muss etabliert werden.

13.2 Bedürfnisse

Verfügbarkeit
Vertraulichkeit
Datenintegrität
Authentizität
Autorisierung
Anonymität
Nichtabstreitbarkeit
Privatheit

13.3 mögliche Bedrohungen

Service Ausfall / Denial of Service
Datendiebstahl / Datenveröffentlichung / Datenerpressung
Datenmanipulation

13.4 mögliche Ursachen

Erdbeben
Feuer
Stromausfall
System- / Ausrüstungsausfall
Terroranschläge
Menschliches Versagen
Viren
Organisierte Unterbrechungen
etc

13.5 Einleitung des Sicherheitsprozesses

- Um einen geeigneten und adäquaten Datensicherheitslevel in einer Organisation zu erreichen und zu halten wird ebenso ein geplanter Ansatz wie auch eine adäquate organisatorische Struktur benötigt.
- Definierung von Sicherheitszielen und einer Strategie um diese zu erreichen.
- Prozess muss von der obersten Stufe des Managements initiiert werden um Wichtigkeit und Konsequenzen klarzumachen.

13.5.1 zu lösende Fragestellungen

Sicherheitsrisiken für die Firma und seine Daten?
Auswirkungen von Sicherheitsproblemen auf kritische Geschäftsabläufe?
Sicherheitsanforderungen durch rechtliche und vertragliche Übereinkünfte?
Standardansätze zur Datensicherheit?
Vorteile von Zertifizierungen?

13.5.2 Risiko- und Schadensanalyse

Beantwortung folgender Fragen:

- Welche Geschäftsprozesse gibt es in der Firma und in welchem Verhältnis stehen sie zu den Geschäftszielen?
- Welche Prozesse hängen von einer funktionierenden IT-Infrastruktur ab?
- Welche Daten sind sind teilweise wichtig und dadurch schützenswert und warum ist dies wichtig? (Kategorisierung)

13.5.3 Richtlinien

siehe oben Kapitel Richtlinien!

13.6 IT Security Officer

Datensicherheit wird oft vernachlässigt und ist normalerweise nebensächlich im alltäglichen Geschäftsablauf.
Wenn die Verantwortlichkeiten nicht klar getrennt werden, wird Datensicherheit schnell zu „jemand anderen sein Problem“, deswegen muss ein ITSO ernannt werden.
Ansprechpartner für alle Datensicherheitsfragen, der ernannt werden sollte um die Datensicherheit zu fördern und zu koordinieren.
Aufgaben:

- Datensicherheitsprozesse kontrollieren und an allen bezüglichen Aufgaben teilhaben.
- Management bei der Erstellung von Richtlinien unterstützen.
- Erstellung des Sicherheits-und Notfallskonzepts und Systemsicherheitsrichtlinien koordinieren.
- Aufstellung zusätzlicher Richtlinien
- Einrichtung und Überwachung von Sicherheitsmaßnahmen.
- Projekte bezüglich der Datensicherheit koordinieren.
- Sicherheitsprobleme untersuchen
- Bewusstseinsteigernde Maßnahmen veranlassen und koordinieren.

13.6.1 Anforderungsprofil

Identifizierung mit den Datensicherheitszielen und Überblick über Aufgaben und Ziele der Firma.
Kooperations- und Teamfähigkeit, aber auch mit guten Durchsetzungsvermögen.
Erfahrung im Projektmanagement.
Unabhängigkeit.

13.6.2 Aufgaben

Datensicherheitsziele und Strategien spezifizieren und Datensicherheitsrichtlinien entwickeln.
Durchsetzung der Richtlinien kontrollieren.
Initiierung, Kontrolle, und Überwachung des Sicherheitsprozesses.
Sicherheitsausbildung und Bewusstseinsteigernde Maßnahmen entwerfen.
IT-Koordinationsteam und Management in Datensicherheitsfragen beraten. Einbindung aller Angestellten in den Sicherheitsprozess.

- Datensicherheit betrifft ohne Ausnahmen alle Angestellten, jeder kann helfen Schaden zu vermeiden und somit zu Erfolg der Firma beitragen.
- Ausbildung anbieten und Bewusstsein steigern.
- Kommunikation, Integration und Meldewege(reporting routes?) unterstützen.

13.7 IT Sicherheitskonzept erstellen

Ziel: pragmatischen und effektiven Ansatz bereitstellen um ein normalen Sicherheitslevel zu erreichen, der auch als Basis für einen höheren Sicherheitslevel dienen kann.

13.7.1 Anwedungsbereich bestimmen

Anwendungsbereich sollte alle Bereiche, Aspekte und Komponenten enthalten, die genutzt werden um spezialisierte Aufgaben, Geschäftsprozesse oder organisatorische Einheiten unterstützen und von intern von der Firma administriert werden.
Wenn das nicht geht, weil z.B. Prozesse von externen Partnern abhängen muss die Schnittstelle klar definiert werden, damit diese ebenfalls in das Sicherheitskonzept aufgenommen werden kann.

13.7.2 Strukturanalyse

Dokumentieren welche Geschäftsprozesse, Anwendungen und Daten im Anwendungsbereich sein sollen.
Netzwerkplan vorbereiten.
IT-System und ähnliche Systeme dokumentieren.
Räume dokumentieren.
Komplexität durch das Formen von Gruppen reduzieren.

13.7.3 Auswahl und Anpassung von Sicherheitsmaßnahmen

mit den Grundschutzbuch arbeiten.

Phase	Typical tasks
Planning and design	• Definition of the intended purpose • Specification of application scenarios • Assessment of the potential risk • Documentation of the decisions made • Drawing up of a security concept • Specification of guidelines for application
Purchasing (if necessary)	• Specification of the requirements regarding the products to be purchased (based on the application scenarios from the planning phase, if possible) • Selection of suitable products
Implementation	• Design and implementation of the test mode • Installation and configuration according to the security policy • Training and sensitisation of all persons involved
Operation	• Security safeguards for current operations (e.g. logging) • Continuous maintenance and further development • Change management • Organisation and implementation of maintenance work • Audit
Disposal (if necessary)	• Withdrawal of authorisations • Deletion of databases and references to this data • Secure disposal of data media
Contingency Planning	• Design and organisation of data backups • Use of redundant equipment to increase the availability • Handling of security incidents • Drawing up a contingency plan

13.8 Desaster

Ein seltenes, natürliches oder menschgemachtes, Ereignis, das die Tätigkeiten einer Gemeinschaft oder eines Betriebes so stört, dass erheblicher und koordinierter Aufwand nötigt ist um eine schnelle Wiederherstellung zu erreichen.

13.8.1 Kontinuitätsplanung

Lernen wie man mit Ereignissen umgeht, die so selten auftreten dass sie gemeinhin nicht als großes Risiko betrachtet werden.
Diese Ereignisse sind riskant da sie ein hohes Schadenspotential besitzen.
Risiko ist das Produkt aus potentiellen Schaden und der Eintrittswahrscheinlichkeit.

13.8.2 Auswirkungen

Verlust an Produktivität, Einkommen, Ansehen und öffentlichem Vertrauen, Wettbewerbsvorteilen.
Verletzung von vertraglichen Bestimmungen
Verletzung von gesetzlichen und behördlichen Bestimmungen.

13.8.3 Plan für Desaster

Keine Verhinderung des Desasters, sondern verhindern von Überraschung und unorganisierter Reaktion.
Organisations- und Meldestrukturen müssen etabliert werden bevor das Desaster die Firma überrascht.

14 Computer Forensics

Generell ist Forensik die Wissenschaft um Gerichtsprozesse zu unterstützen.
Computerforensik bedeutet:

- Daten von Speichermedien oder Netzwerkprotokollen sicherstellen
- Überprüfung von Computersystemen
- Meinungsbericht wiedergeben

Die gleichen Technologien werden auch für Industriespionage benutzt.
Computerforensik ist die Suche nach Beweisen, die auf digitalen Geräten gefunden werden können die in einem Zwischenfall verwickelt waren.

14.1 Was ist es nicht?

Proaktiv:

- reagiert auf Ereignisse oder Anfragen

Das Finden des „bösen Jungen“:

- Es geht um das finden von wertvollen Beweisen.

14.2 Schritte

Herunterfahren

- Einige Daten sind eventuell flüchtig im Speicher gespeichert - versuchen diese zu retten.

Konfiguration dokumentieren

- angeschlossene Geräte/Netzwerke
- Verkabelung

Daten duplizieren

- Image (1-zu-1 Kopie) der gespeicherten Medien erstellen um Manipulation der Beweismittel zu verhindern.

- Jegliche Änderungen an den duplizierten Daten verhindern.

Daten sortieren
Daten wiederherstellen

- Clean room technology (?)

14.3 Zerstörung von Daten

Schredder
Mehrfaches Schreiben von Zufallsdaten
Magnetische Felder

14.4 Absichten/Ziele

Computersysteme analysieren
Daten wiederherstellen im Falle eines Defektes
Computersystem nach einem Angriff analysieren

- herausfinden wie der Angreifer Zugriff erlangen konnte
- herausfinden was der Angreifer getan hat

Beweise gegen einen Angestellten sammeln den die Firma kündigen möchte.
Informationen darüber gewinnen wie Computersysteme funktionieren (reverse Engineering)

14.5 Phasen

Systemkonservierung, Beweissuche, Ereignisrekonstruktion

14.5.1 Systemkonservierung

Ziel: Menge der Beweise die evtl überschrieben werden reduzieren/verhindern

- Live Systeme müssen Daten aus flüchtigen Speichern bewahren
- Tote Systeme enthalten Daten in nicht flüchtigen Speichern

Typische Aktionen:

- Speicherauszug (Memory dump) erstellen
- Image der originalen Daten erstellen

14.5.2 Beweissuche

Suchen ist ein recht einfacher Prozess

- Eigenschaften des Objektes definieren
- nach dem Objekt in einer Sammlung von Daten suchen

Wenn die Sammlung groß ist muss Ermittler wissen wo er suchen muss

- verschiedene Speichermedien
- Verschiedene Orte auf den Medien
- Adressen filtern (IP-Adressen, Port, Herkunft)
- Nach Passwörtern filtern

14.5.3 Rekonstruktion

Daten analysieren um herauszufinden welche Ereignisse in dem System stattgefunden haben.
Versuchen solche Fragen zu beantworten:

- Wem gehört eine Datei?
- Wer hat einen Account erstellt
- Wann ist das System zusammengebrochen?
- Wie schnell war das Auto?

Bezug zwischen digital gespeicherten Ereignissen und physischen Ereignissen herstellen.

14.6 Datenanalyse

14.6.1 Analyse von physischen Speichermedien

Gerät welches Daten speichert
typischerweise in Blöcken organisiert
BSP: HDD, USB-Sticks, CD/DVD/Bluray....

14.6.2 Datenträgeranalyse

Speicherbereich für ein einzelnes Dateisystem
Kann sich über mehrere Medien spannen.
befindet sich normalerweise in einer Partition
auch bekannt als „logisches Laufwerk“

14.6.3 Dateisystemanalyse

Enthält Dateien und Verzeichnisse
Enthält ebenso Metadaten

- Zeiten
- Zugriffsrechte
- Protokolle

Für den Nutzer direkt über das Betriebssystem verfügbar.

14.7 Probleme bei der Untersuchung von Festplatten

Verschlüsselung
Unbekannte Dateisysteme
Versteckte Daten
Interne Konfigurationen der Festplatte (z.B. Dead Sector Mapping)

14.8 Achtung!

Vieles weggelassen was einzelne Methoden, Dateisysteme etc angeht!

15 Gebäudesicherheit

Funktionen von Gebäuden bei der IT-Sicherheit:

- Einschränkung des Zugangs zu Rechnersystemen
- Sichere Beherbergung der Rechnertechnik
- Bereitstellung der notwendigen Infrastruktur (Strom, Datenleitungen, etc...)

Gefährdungen:

- Höhere Gewalt (Wasser, Feuer, Sturm, Blitz)

- Menschliches Versagen (Fehlbedienungen, Verkehrsunfälle...)
- Technisches Versagen (Ausfall der Versorgungsleitungen (Strom/Daten) oder Sicherungseinrichtungen)
- Vorsätzliche Handlungen (unbefugtes Eindringen, Zerstörung, Diebstahl, Vandalismus, Anschläge)

Maßnahmen:

- Einbruchschutz (Zaun, Schließsysteme, Alarmanlage etc.)
- Schutz vor Ausfällen und höherer Gewalt (Standort, Schutzeinrichtungen, Feuerlöscher etc.)

15.1 Zugriffskontrolle

Mechanische Schlüssel
Biometrie
Elektronische Schlüssel / Karten
Mechatronische Schlüssel
Pförtner

15.1.1 Gründe für mechatronische Schlüssel

konventionelle Schlüssel sind unsicher und unhandlich

- Duplikate von Schlüsseln
- Keine Audit-Informationen(?)
- keine zeitliche Beschränkung
- Generalschlüssel
- Schlüsselauslieferung umständlich
- Keine Rückziehung von Schlüsseln möglich, umständlicher Schlossaustausch

15.1.2 Angriffe

Gewaltsamer Zugriff:

- Mechanische Zerstörung des Schlosses
- Standards verlangen mind. 5 Min Widerstandszeit

Verdeckter Zutritt:

- Zutritt ohne das Schloss zu zerstören und ohne Spuren zu hinterlassen (Picking, Generalschlüssel)
- Standards verlangen 10-15 Minuten Widerstandszeit

Angriff mit Schlüssel:

- Angriff auf Schlüsselverwaltung (Duplikat erstellen, Schlüssel nachbestellen)
- Gute Voraussetzungen: Verfügbarkeit von Blanko-Schlüsseln oder Generalschlüsseln

15.1.3 Funktionen mechatronischer Schlösser

Zugang:

- Schlüssel und Schloss authentisieren sich gegenseitig (Public Key Verfahren)
- Schloss hat änderbare Liste von erlaubten Schlüsseln oder Schlüssel-IDs oder Public Keys der Schlüssel
- Schloss öffnet
- Schloss trägt Vorgang ins Log ein

Zeitlich beschränkter Zutritt:

- Schlüssel und Schloss authentisieren sich gegenseitig

  - Public Key Verfahren
  - Schlüssel muss die aktuelle Uhrzeit liefern. Schlüssel und Schloss entscheiden anhand der Uhrzeit über Zugang
  - Challenge-Response um Man-in-the-Middle mit Zeitversatz zu verhindern.

- Schloss öffnet
- Schloss trägt Vorgang ins Log ein

Programmierung:

- Software liefert Liste der erlaubten Kombinationen aus Schlüsseln und Schlössern über Programmiergerät an Programmierschlüssel.
- Programmierschlüssel merkt sich Listen
- Programmierschlüssel und Schloss authentisieren sich gegenseitig
- Schloss enthält Liste gültiger Programmierschlüsseln oder kennt Liste von gültigen root-Zertifikaten
- Schloss empfängt Liste von erlaubten Schlüsseln und speichert diese
- Schloss überträgt Log-Files an den Programmierschlüssel

16 Cloud Security

Großer Hype
Viele kümmern sich nicht genug um Sicherheit und Privatheit
Andere kümmern sich zu sehr darum. Private Cloud:

- Nur ein neuer Weg um Ressourcen zuzuweisen
- Keine neuen Angriffe

Public Cloud:

- Das was gemeinhin als cloud bekannt ist, also Google und co bieten Rechenleistung an

hybride Cloud:

- besteht zum Teil aus privater/interner Cloud und zum anderen Teil aus öffentlicher/externer Cloud.

16.1 Bedrohungen

Standardanforderung:

- Verfügbarkeit und Zuverlässigkeit
- Privatheit
- Vertraulichkeit
- Integrität
- Authentizität

besondere Herausforderungen mit Cloudcomputing:

- Angriffe auf Privatheit von Daten
- Host Level Attacks
- für virtualisierte Computer spezifizierte Angriffe. Bedrohung für den Hypervisor
- Angriffe auf die Infrastrukur
- Angriffe auf Anwendungen
- Angriffe auf Persönlichkeiten

16.2 Lösungen

Schutz der Technik und deren Fortbestand (Ausfallstrategien, Redundanz)
Kontrolle der Technik (Netzwerkschnittstellensicherheit, Browsersicherheit, Schlüsselverwaltung)
Nutzerverwaltung
Sicherheitsmanagement

17 Human resources security

Awareness:

- Was ist das Problem?
- kann mit Zeitungen, Postern, Videos beigebracht werden

Training:

- Wie können wir das Problem lösen?
- Mit Vorlesungen, workshops zu lehren.

Education:

- Warum gibt es das Problem?
- Kann mit Dikussionen, Hintergrundliteratur beigebracht werden.

Sicherheit im Einstellungsprozess:

- Hintergrundchecks und Sicherheitsuntersuchungen
- Detaillierte Mitarbeitervereinbarungen mit Verpflichtungen bzgl Sicherheit, Vertaulichkeit und Geheimhaltung.

Sicherheit währen der Anstellung:

- niedrigst niedrige Privilegierungslevel vergeben
- Teilung der Pflichten und cross checks
- begrenzte Abhängigkeit von Schlüsselangestellten (Bus-Faktor)

Kündigung:

- Löschen von Autorisierungslisten
- Zugriff auf Codes entfernen
- Schlösser einsammeln
- Firmenbesitz zurückholen
- Geschäftspartner und Abteilungen über Kündigung unterrichten (damit der nicht anruft und im Namen der Firma handelt)

18 Software Assessment

18.1 Code Revision vs Black Box Penetration Testing

White Box Test:

- Prüfer arbeitet mit dem Quelltext
- manuelle Code Inspizierung
- automatisierte Tools wie RATS, ITS4, Pathfinder....
- komplette Codeabdeckung ist möglich
- Tools sind nicht perfekt und benötigen manuelle Kontrolle
- Manchmal fehlt es an Zugriff auf den Quellcode

Black Box Testing:

- Prüfer „füttert“ das System mit Eingaben
- ist immer möglich
- Portierbarkeit: Man kann verschiedene Anwendungen mit der gleichen Testsuit testen
- Einfach
- schlechte Abdeckung
- manuelles Testen oder Fuzzing (automatisch generierte zufällige Eingaben für das System)

Gray Box Testing:

- kombiniert Black Box Testing mit Reverse Engineering (RE)
- RE wird genutzt um eventuell verwundbare Stellen zu finden
- Manuelles Gray Box Testing
  - Nutze Tool (IDA PRO o.ä.) um aus Binärcode, Assembler-Code zu erzeugen
  - mögliche Schwächen identifizieren
  - Probeeingaben erstellen
- automatisiertes Gray Box Testing
  - Mehrere Tools um den Prozess zu automatisieren (BugScam, Inspector, Bin Audit)
- Pro:
  - Kann immer gemacht werden
  - bessere Abdeckung als Black Box Testing
- Contra:
  - RE ist sehr schwierig

18.2 Vertrauensbeziehungen

Verschiedene Rechner in einem System setzen unterschiedliche Grade an Vertrauen ineinander
Vertrauensbeziehungen müssen eindeutig und gut erforscht werden.
Transitives Wesen von Vertrauen

18.2.1 Unangebrachtes Vertrauen

Eine unbegründete Annahme machen.
Eingabe:

- Die meisten Schwachstellen werden durch bösartige Eingaben ausgelöst
  - Entwickler nimmt an das niemand eine Telephonnummer mit mehr als 5000 Zeichen eingibt

Schnittstellen:

- Entwickler nutzt eine Schnittstelle die nicht genügend Schutz vor Angriffen von außen bereitstellt.
- falsche Konfigurierung von Schnittstellen
- Annahme dass der Zugriff auf eine Schnittstelle zu schwierig ist für einen Angreifer

Umgebung:

- Software läuft nicht im Vakuum
- Entwickler vertraut der Umgebung, aber der der Angreifer kann diese manipulieren

Ausnahmen:

- Angreifer verursacht unerwartete Änderungen im Anwendungsablauf durch äußere Maßnahmen.

18.3 Design Review

Algorithmen können unter Umständen Angriffsflächen bieten
Vertrauensbeziehungen:

- Vertrauensgrenzen
  - spiegelt die Eingrenzung des Vertrauens zwischen Modulen wieder
- Vertrauensquellen
  - Regionen des geteilten Vertrauens, eingeschränkt durch Vertrauensgrenzen
- Vertrauensmodell
  - Abstraktion die diese Beziehungen repräsentiert

18.3.1 Bedrohungsmodellierung

Prozess während der Designphase der in späteren Entwicklungsphasen aktualisiert wird.

- Informationssammlung (Anwendung verstehen)
- Modellierung der Anwendungsarchitektur (Datenflussdiagramm entwerfen)
- Bedrohungsidentifikation (Angriffsbäume verwenden)
- Dokumentation der Funde
- Priorisierung des Implementationsreviews

19 Timing

Kryptografische Nutzung von Uhren:

- Schlüsselmanagement (gültig von bis)
- Reihenfolge von Ereignissen in verteilten Systemen

19.1 Ablaufzeit

Wir möchten die Gültigkeitsdauer von Dokumenten beschränken (Flugtickets, Copyrights)

- Ablaufzeit ist normalerweise im Dokument enthalten
- aktuelle Zeit muss bekannt sein

Eine Uhr stellt einen für eine Maschine einzigartigen Wert bereit (Wert der niemals in einem definierten Bereich doppelt verwendet wird)
Bereich kann sein:

- Session
- Zwischen zwei reboots
- Lebensdauer des Rechners

Nonces (vorläufige Buchstaben oder Zahlenfolg laut wiki) werden für verschiedene kryptografische Prozesse genutzt (Pseudozahlengenerator, Time Codes)

19.2 Monotonie

- Nutzen die Eigenschaft der Zeit dass sie immer vorwärts läuft
  - Einbindung von Zeit in ein kryptografisches Protokoll verhindert das Angreifer versuchen können ältere Nachrichten abzufangen
- Prüfen und Loggen
  - Wenn es Konflikte gibt, liefert der Log Daten um die genaue Abfolge von Ereignissen nachzuverfolgen

19.3 Echtzeittransaktionen

- Uhren müssen zuverlässig sein
  - Neue Transaktionen dürfen keinen Zeitstempel haben der geringer ist als die aktuelle Zeit

19.4 Sicherheitsgefahren

- Zeit zurück setzen
- Zeit anhalten
- Uhr vorstellen

19.5 Zuverlässige Uhr erstellen

mögliche Eingaben:

- CPU Taktzyklen
- letztes Runterfahren des System verfolgen
- NTP/SNTP - authenticated NTP

Ptobleme:

- Zeitzonen
- Schaltsekunden

19.6 Same State Problem

Jeder Rechner ohne Echtzeituhr startet bei jedem Bootvorgang in exakt demselben Zustand

- Liest dieselben Programme
- initalisiert dieselbe Hardware
- Setzt denselben Startpunkt für den Zufallsgenerator
- Wird dieselben Challenges in einem Challenge-Response Schema generieren

Lösungen:

- Echtzeituhr
  - Probleme:
  - Manipulation
  - Strom
- Rechner neustarten
  - Probleme:
  - Nichtflüchtiger aber beschreibbarer Speicher wird benötigt
  - Manipulation
  - Manche NVRAM sind beschränkt in der Menge der Schreiboperationen