Absicherung von Netzübergängen

Teil 1: Kommunikationssicherheit

Sicherungsprotokolle auf der

- Datensicherungsschicht (Schicht 2)
- Netzwerkschicht (Schicht 3) und
- Transportsschicht (Schicht 4)

Teil 2: Absicherung von Netzübergängen

Sicherheitskomponenten und Infrastrukturen

- Paketfilter, Firewalls, Intrusion Detection/Prevention
- Firewall Architekturen
- Firewall Organisation und Netzsicherheitskonzepte

1 Firewalls

Netzsicherheit: Worum geht es?

- Ein Unternehmen hat den Bedarf, die eigene IT sicher zu betreiben und gleichzeitig über das Internet erreichbar zu sein bzw. Internet-Ressourcen zu nutzen.
- Das interne Netz wird häufig als sicher angenommen.
  - Zentrale Kontrolle, geeignete organisatorische und personelle Maßnahmen.
- Das Internet ist prinzipiell unsicher
  - Offene Architektur, keine zentrale Kontrolle, anonyme Zugriffe

Am Netzübergang zwischen Netzen unterschiedlicher Vertrauenswürdigkeit

- Netze weisen ein unterschiedliches Sicherheitsniveau auf (z. B. Internet vs. Intranet)
- Netze stehen unter der Kontrolle verschiedener Organisationen (z. B. Kopplung von Unternehmensnetzen)

Mögliche Schutzansätze:

Ansatz 1: Sichere Systeme und Anwendungen

- Sämtliche Rechner im internen Netz werden sicher konfiguriert und betrieben.
  - Es gibt viele bekannte Sicherheitslücken
  - Bereits bei geringer Rechnerzahl sehr aufwändig und fehleranfällig

Ansatz 2: Netzsicherheit durch eine Firewall

- Konzentration des Netzverkehrs zwischen dem Internet und dem internen Netz auf ein zentrales System (= Firewall), das den Verkehr kontrolliert. (Pförtnerfunktion)
- Vorteile:
  - Die Firewall überwacht und filtert den Netzverkehr.
  - Fehlkonfigurationen interner Systeme gefährden nicht sofort die Gesamtsicherheit.
  - Netzzugriffe auf kritische interne Systeme/Applikationen sind steuerbar.
  - Aufwandsminderung, da „nur“ Firewall sicher konfiguriert werden muss
    => geringeres Risiko von Fehlkonfigurationen

Definition: Firewall-System

- Ein Firewall-System ist ein System (ein oder mehrere IT-Systeme), das den Netzverkehr zwischen zwei Netzen zentral kontrolliert und regelt.
- Durch Firewall-Systeme werden typischerweise Netze unterschiedlicher Vertrauenswürdigkeit verbundenen
- Die Firewall regelt den Verkehr, d. h. sie bestimmt welcher Netzverkehr in welche Richtung passieren darf. (FW setzt gegebene Netzwerk-Policy um.)

Grenzen von Firewalls, kritische Punkte:

- Konfigurationsfehler in der Firewall können schwerwiegende Auswirkungen haben auf
  - die Sicherheit interner Systeme,
  - die Erreichbarkeit interner Server aus anderen Netzen (z.B. Internet),
  - die Erreichbarkeit des Internets für interne Hosts.
- Es gibt verschiedene FW-Typen, die jeweils eine spezielle Kontroll- und Filterfunktionalität aufweisen.
- Firewalls selbst sind i.d.R. Softwaresysteme (ggf. als Applicance ausgeführt) und können daher SW-Schwachstellen enthalten.
- Angreifer können versuchen, unzulässigen Netzverkehr in Netzverkehr zu verstecken/zu tunneln, der von der FW durchgelassen wird.
- Nutzer können versuchen Wege zu finden, um die Restriktionen der Firewall zu umgehen. (Z.B. Installation eines DSL-Routers.)

Firewall Typen:

- Paketfilter (statisch und dynamisch)
  - Kontrolle auf der Netzwerkschicht
  - Kontrolle auf Basis von Headerinformatinonen (IP-Header, TCP/UDP-Header)
  - Stateful inspection = dynamische Paketfilterung
- Deep Inspection
  - Deep Inspection = Übertragene Anwendungsdaten werden mit analysiert
  - Der Begriff Deep Inspection wird sowohl beim Intrusion Detection verwendet, als auch bei Firewalls: Deep Inspection Firewall
- Content Filtering
  - Inhaltliche Prüfung / Filterung von Anwendungsdaten
  - Einsatz z.B.: Web Filter Firewall
  - Sperrung vom Unternehmen unerwünschter Web-Seiten (Social media, Porno, …)
  - Ausgehende Prüfung ggf. zur Kontrolle, dass bestimmte Inhalte das Unternehmen nicht über das Netz verlassen (Data Loss Prevention)
- Proxy Firewalls (Proxy = Stellvertreter)
  - Proxy auf Transportschicht => Circuit Level Gateway
  - Proxy auf Anwendungsschicht => Application Level Gateway, Application Proxy
- Personal Firewall, Endpoint Firewall
  - Kontrolliert Netzverkehr auf den Endsystemen (vergleiche HIDS / HIPS)
- Kombinierte Firewalls (Next-Generation Firewalls), kombinierte Funktionalität aus
  - Stateful inspection, ggf. Application Proxies,
  - Integrierte Intrusion Detection und Intrusion Prevention Funktionen,
  - Ggf. applikationsabhängige Deep Packet Inspection und Content Filterung,
  - Upgrade Funktionen um zukünftige Erweiterungen/Entwicklungen zu berücksichtigen.

Unterscheidung danach, WAS und WIE geprüft wird:

- Erfolgt die Prüfung zustandsbasiert (statefull, dynamisch) oder zustandslos (stateless, statisch)?
- Werden im Wesentlichen Headerinformationen analysiert? (=> Paketfilterung)
- Werden Anwendungsdaten analysiert? (=> Deep Inspection, Content Filerterung)

Unterscheidung danach, WO und wie transparent geprüft wird:

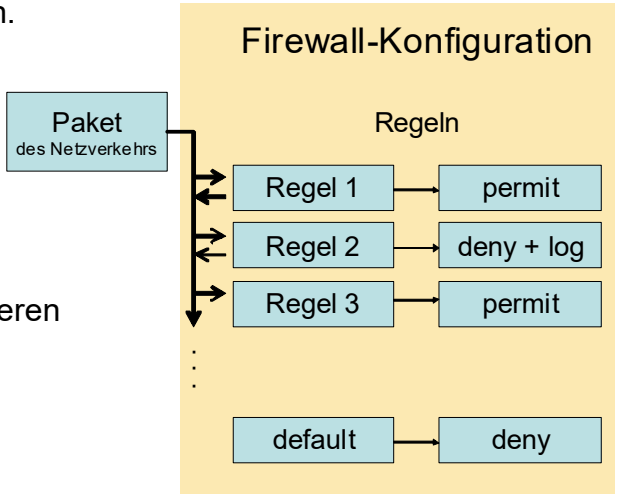
- Lediglich Abgriff des Netzverkehrs (TAP, SPAN): Lediglich zu Analyse (=> Intrusion Detection), keine Blockierung möglich
- Inline Transparent: Pakete werden blockiert oder durchgelassen
- Inline als Proxy: Pakete werden angenommen und neu aufgebaut
- Auf dem Endsystem (=> Personal Firewall, Endpoint Firewall)

TAP=Test Access Point, SPAN=Switch Port Analyzer

Paketfiltern: Funktionsweise und Konfiguration

- Konfigurationsgrundsatz:
  - Default deny
  - Alles, was nicht explizit erlaub ist, ist verboten.
- Paketfilter kontrollieren den Verkehr auf der Basis von Regeln
  - Paketfilter-Konfiguration = Regelwerk
  - Regel werden der Reihe nach durchlaufen
  - Sobald eine Regel gefunden wird, die auf das zu beurteilende Paket passt, erfolgt die für diese Regel vorgesehene Aktion. Die weiteren Regeln werden dann nicht mehr durchlaufen.

  => Reihenfolge der Regeln ist relevant!



- Aktionsarten
  - Weiterleitung des Pakets (permit)
  - Verwerfen des Pakets (deny)
  - Aufzeichnung des Pakets (log)
  - Ggf. Senden einer Fehlermeldung oder eines Alarms an ein anderes System

Statische Paketfilterung (Stateless Packet filtering FW)

- Filterung auf der Ebene der Netzwerkschicht.
- Effiziente und transparente Filterung auf der Basis von IP-Paket en
- Entscheidungsrelevante Informationen auf IP-Paketebene:
  - IP-Quelladresse (IP-Header)
  - IP-Zieladresse (IP-Header)
  - Transportprotokoll (IP-Header)
  - Bei TCP:
    - » TCP Quellport (TCP-Header)
    - » TCP Zielport (TCP-Header)
    - » ACK-Bit (TCP-Header, in allen PDUs außer der ersten initialen PDU gesetzt)
    - » SYN-Bit (TCP-Header, nur in den ersten beiden PDUs gesetzt)
  - ggf. Optionen (IP-Header, Extension Header bei IPv6)
  - ggf. ICMP Meldungsart
  - bei UDP: Quell- und Zielport
  - Netzinterface (NIC) auf dem das Paket ankommt
- Typische Realisierung: Router mit Access Control List (ACL)

Beispiel: Filterung der E-Mail Kommunikation

- E-Mail Kommunikation zwischen zwei SMTP-Servern über einen Paketfilter
  - externer SMTP-Server an NIC1 des Paketfilters,
    Server steht irgendwo im Internet: IP-Netzmaske „Any“ (= beliebig)
  - interner SMTP-Server an NIC2 des Paketfilters, IP-Adresse ServerAdr
- Sendender Server baut TCP-Verbindung zu TCP-Zielport 25 auf, wobei er einen TCP-Quellport > 1023 verwendet.

Konfiguration: Für jede TCP-Verbindung sind 2 Regeln erforderlich

- Regel 1 und 2: Passieren eingehender E-Mails
- Regel 3 und 4: Passieren ausgehender E-Mails
- Regel 5: Default Deny
- 3 Wege Handschake TCP-Verbindungsaufbau. Flags: 1. SYN, 2. SYN/ACK, 3. ACK

Segmente zur Initiierung einer TCP-Verbindung sind an nicht gesetztem ACK-Flag erkennbar!

*=eingehend über diese NIC (Network Interface Card)

Rule	NIC*	IP-Src	IP-Dest	Protocol	SrcPort	DestPort	ACK	Action
1	NIC1	Any	ServAdr	TCP	>1023	25	Any	Permit
2	NIC2	ServAdr	Any	TCP	25	>1023	Yes	Permit
3	NIC2	ServAdr	Any	TCP	>1023	25	Any	Permit
4	NIC1	Any	ServAdr	TCP	25	>1023	Yes	Permit
5	Any	Any	Any	Any	Any	Any	Any	Deny

Schutzwirkung:

- Es wird speziell der Verkehr mit dem SMTP-Server zugelassen. Verkehr an andere IP-Adressen wird blockiert.
- Anderer Verkehr als SMTP wird blockiert.
- Durch Filterung des ACK-Flags werden Verbindungsaufbauten nur in die definierte Richtung zugelassen.

Empfehlungen zur Filterung am Übergang zum Internet

- DNS nur für dedizierte interne Name Server freigeben
- ICMP-Pakete ausgehend sperren: Keine Ping-Antworten (Echo-Reply) ins Internet
- IP-Pakete mit „Source-Routing“ Option sperren
- Windows-Ports sperren (Netbios, 137, 138, 139)
- Generell: Default deny Prinzip

Diskussion statische Paketfilterung:

- Vorteile:
  - Einfach, effizient, Hardware-nahe Devices
  - Kontrolle ist für die Dienste transparent
- Nachteile statischer Paketfilter
  - In die Regeln gehen nur wenige Informationen ein.
  - Nutzdaten von Applikationen werden nicht untersucht.
  - Ein statischer Paketfilter ist zustandslos. Bei Abhängigkeiten zwischen Paketen kann nicht immer zwischen berechtigten und unberechtigten Paketen unterschieden werden. (Angriff über fragmentierte Pakete)
  - Pakete werden lediglich anhand ihrer IP-Adressen und Portnummern identifiziert. Insbesondere erfolgt keine Prüfung, ob es sich tatsächlich um ein Paket der angegebenen Form handelt.
  - Da Clients Portnummern zufällig >1023 wählen, müssen große Portbereiche freigeschaltet werden, um die üblichen Dienste nutzen zu können.
  - Definition der Filterregeln ist aufwändig und damit häufig fehlerbehaftet.
  - Filterregeln sind schwer zu warten.
  - Die Pakete werden lediglich anhand der IP-Adressen identifiziert. Eine Authentifizierung erfolgt nicht. IP-Spoofing ist möglich.
  - Keine Nutzer-spezifischen Logging-Informationen.

Beispiele für Angriffe bei statischer Paketfilterung:

- Fragmentierte IP-Pakete
  - Selbst bei Blockieren sämtlicher eingehender Verbindungsaufbaupakete (SYN=1, ACK=0) kann ein solches Paket bei geschickter Fragmentierung erzeugt werden.
- SYN-Flooding: Denial-of-Service Angriffe
- IP-Spoofing: Maskerade durch Fälschen von IP-Adressen

Dynamische Paketfilterung (zusandsbasiert, stateful inscpection FW)

Beispiele für Zustandsdaten:

- Zustand einer TCP-Verbindung (bisherige TCP-Flags)
- Ausgetauschte Portinformationen (z. B. bei FTP, RPC, UDP)
- Daten über die Fragmentierung von Paketen
- ggf. Datenfluss zwischen Applikationen => Deep Packet Inspection

Vorteile der stateful inscpection Paketfilterung gegenüber statische Paketfilterung

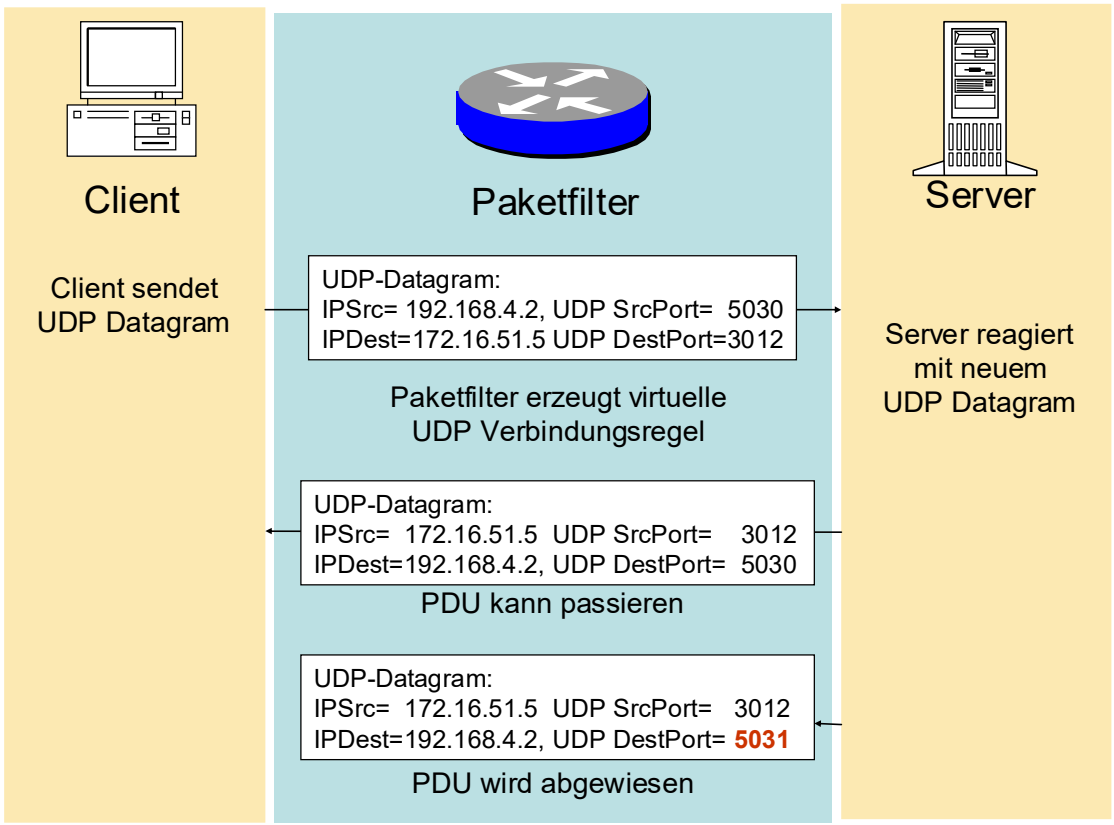
- Zahlreiche Regeln werden temporär durch den Paketfilter selbst gebildet. Daher sind zur Konfiguration weniger Regeln erforderlich. Die Konfiguration wird dadurch übersichtlicher und die Wahrscheinlichkeit von Konfigurationsfehlern wird verringert.
- Die temporären Regeln werden auf Grundlage bereits beobachteter Pakete erstellt. Die temporären Regeln enthalten daher exakte Angaben zu IP-Adressen und Portnummern und bieten so einen höheren Schutz, als die Regeln bei statischen Paketfiltern.

Umgang mit TCP-Diensten

- TCP ist verbindungsorientiert, 3-Wege-Handshake zum Verbindungsaufbau
- Paketfilter kann Pakete mit Anfragen für den Verbindungsaufbau und Annahmen eines Verbindungsaufbaus identifizieren:
  - 1. Paket: Verbindungsaufbau-Anfrage: SYN-Flag gesetzt, ACK-Bit nicht gesetzt
  - 2. Paket: Verbindungsaufbau-Annahme: SYN-Flag gesetzt, ACK-Bit gesetzt
  - Alle weiteren Pakete: SYN-Flag nicht gesetzt, ACK-Bit gesetzt
- Nur über Verbindungsaufbau-Anfragen können Verbindungen aufgebaut werden!
  - Andere Pakete sind sicherheitstechnisch weniger relevant, da über sie keine neuen Verbindungen aufgebaut werden können.
- Statische Regel für Verbindungsaufbau-Anfragen
  - Aus Paket, auf das die Regel zutrifft, wird temporäre Regel für Rückrichtung gebildet
    => Vertauschen von IP-Adressen und Portnummern, ACK = YES (muss gesetzt sein)
  - Temporäre Regel wird nach Verbindungsabbau bzw. Timeout wieder gelöscht.
- (Genauere Kontrolle ist prinzipiell möglich über die Verfolgung der aktuellen Zustände im TCP Zustandsgrafen.)

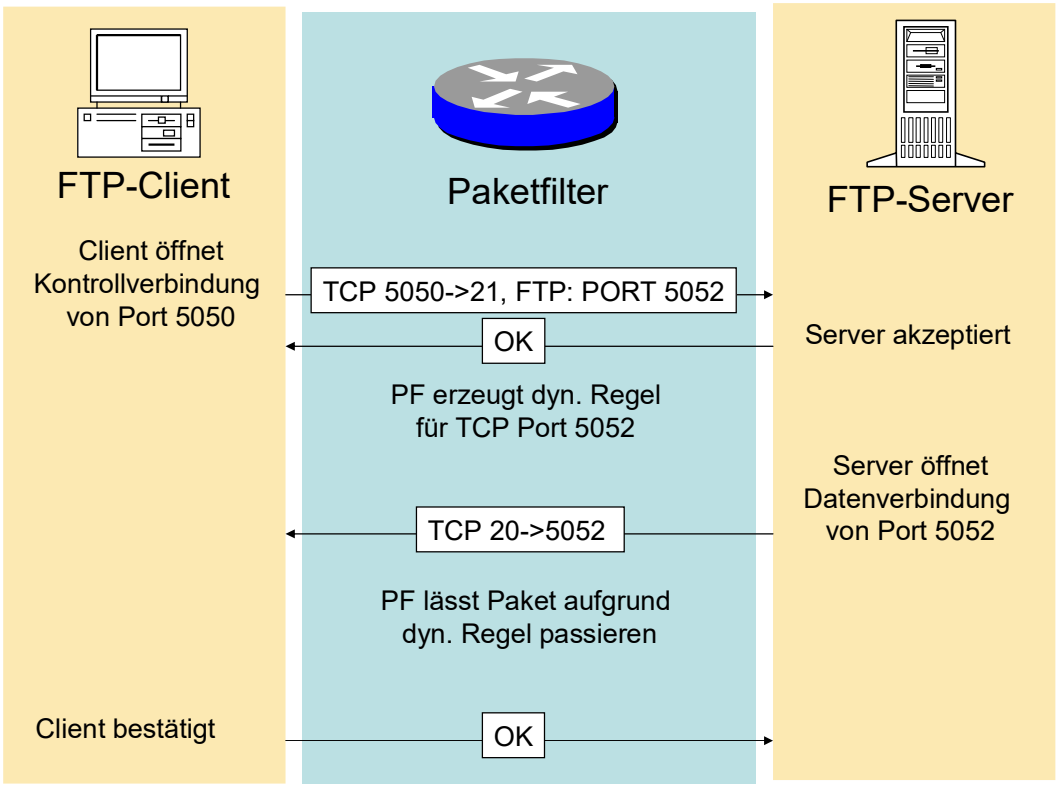
Beispiel: Umgang mit UDP-Diensten:

- UDP (User Datagram Protocol) ist verbindungslos
  - Bsp.: Domain Name Service (DNS), Network Time Protocol (NTP)
- Problem:
  - Unterscheidung zwischen Anfrage und Antwort ist nicht direkt möglich
  - Statische Paketfilter können Richtung des Dienstes nicht kontrollieren
- Lösung durch dynamische Paketfilter:
  - Für UDP-Pakete, die gemäß Filterregeln durchgelassen werden, wird eine virtuelle Verbindung (IP-Adressen, UDP-Portnummern) hergestellt.
  - Danach werden UDP-Pakete durchgelassen, wenn sie zur virtuellen Verbindung gehören.
  - Virtuelle Verbindung wird nach konfigurierbarer Zeit der Inaktivität geschlossen.



Umgang mit FTP:

- FTP ist TCP-basiert. Ablauf (Active FTP):
  - Client baut Kontrollverbindung (KV) zum Server auf (Port 21).
  - Server baut für jede Dateiübertragung eine Datenverbindung (DV) zum Client auf (Port 20).
  - Client Portnummer ist dynamisch (>1023). Der Client teilt dem Server über die KV seine Portnummer für die DV mit (FTP Kommando PORT)
- Statischer Paketfilter kann nicht zwischen verschiedenen Clientports unterscheiden.
- Dynamischer Paketfilter:
  - Untersucht FTP-Nutzdaten und ermittelt vom Client genutzte Portnummer
  - Nur dieser Port wird in Richtung des Clients für den Aufbau von FTP-Datenverbindungen geöffnet.
  - Temporäre Regel wird erzeugt.
  - Regel wird mit Abbau der Kontrollverbindung wieder gelöscht (FTP Kommando QUIT)



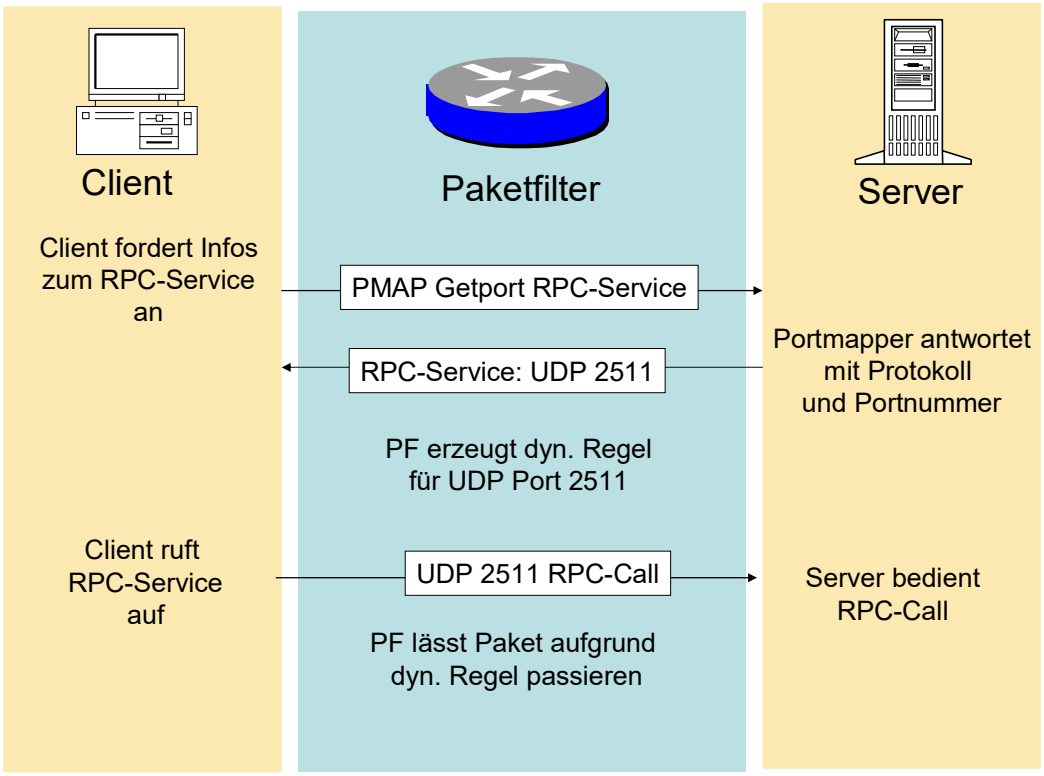
Empfehlung für ausgehende FTP-Zugriffe (interne FTP-Client): Passives FTP

- Client schickt PASV Kommando, Server schickt DV-Portnummer für Annahme DV
- Client baut dann DV von >1023 Port zur DV-Portnummer vom Server auf

=> kein Verbindungsaufbau von extern Server zu internen Clients erforderlich

Umgang mit RPC-Diensten (Remote Procedure Call)

- Funktionsweise:
  - RPC-Clients fragen konkrete RPC Portnummern über den Portmapper-Service ab.
  - Portmapper-Service UDP und TCP Port 111
- Problem:
  - RPC-Dienste haben (i.d.R.) keine festen Portnummern
- Lösung:
  - RPC GETPORT Anfragen von RPC-Clients an den Portmapper-Service und die zugehörigen Antworten, die die konkreten UDP Portnummern enthalten, werden kontrolliert.
  - Für die RPC-Portnummern werden dynamische Filterregeln erstellt (virtuelle RPC-Verbindung)
  - RPC-Pakete, die nicht an den Portmapper gehen, werden nur durchgelassen, wenn sie zu einer virtuellen RPC-Verbindung gehören
  - Virtuelle Verbindung wird nach konfigurierbarer Zeit der Inaktivität wieder geschlossen.

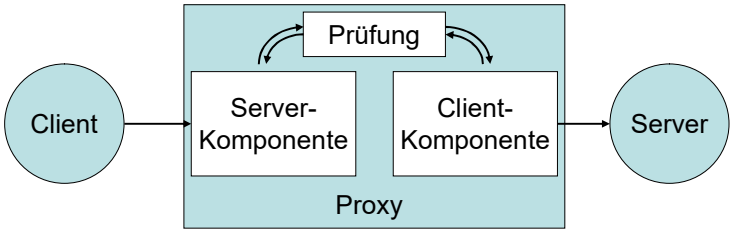


„Hole Punching" Angriff auf Stateful inspection Firewalls

- Ziel: Eine interne Applikationen „will" aus dem Internet über Ports erreichbar sein, die auf der Firewall in eingehende Richtung (extern => intern) gesperrt sind
- Zugriffe in ausgehende Richtung (intern => extern) sind i.d.R. erlaubt
- Angriff: Die interne Applikation sendet regelmäßig Pakete an die Firewall, so dass temporäre Freigabe-Regeln für die Rückrichtung ständig neu erzeugt werden.

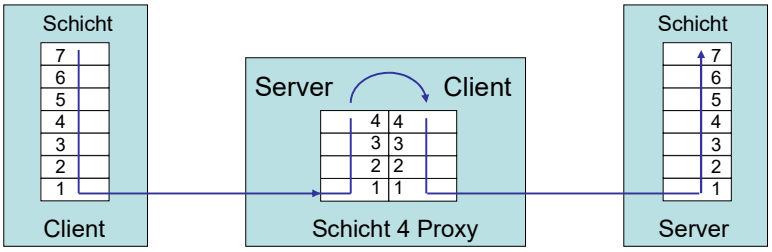
Proxy-Firewalls

- Proxy = Stellvertreter, Vermittler
- Proxy tritt gegenüber Clients als Server auf und gegenüber dem Server als Client
- Clients bauen eine Verbindung zum Proxy auf, dieser baut dann eine Verbindung zum Zielserver auf.
- Der Proxy kann den Netzverkehr zusätzlich kontrollieren



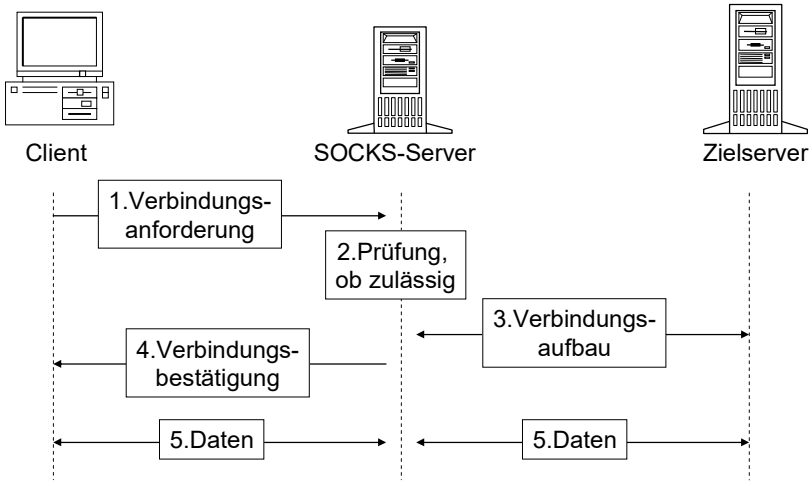
Proxy auf Transportschicht: Circuit Level Gateways

- Beim Circuit Level Gateway kopiert der Proxy die Transportschicht-Nutzlasten (TCP, UDP) von einem IP-Stack (Client) in den anderen (Server).



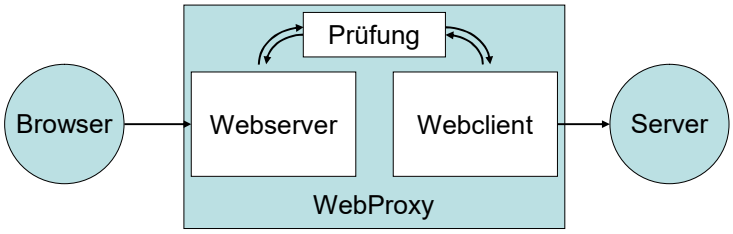
SOCKS

- am weitesten verbreitete Realisierung eines Relay-Servers
- spezifiziert in RFC 1928, Referenzimplementierungen auf gängigen Windows/Unix-Systemen verfügbar
- SOCKS ist vollständig unabhängig von der Applikation und übernimmt nur die „Vermittlung“ zwischen Client und Server

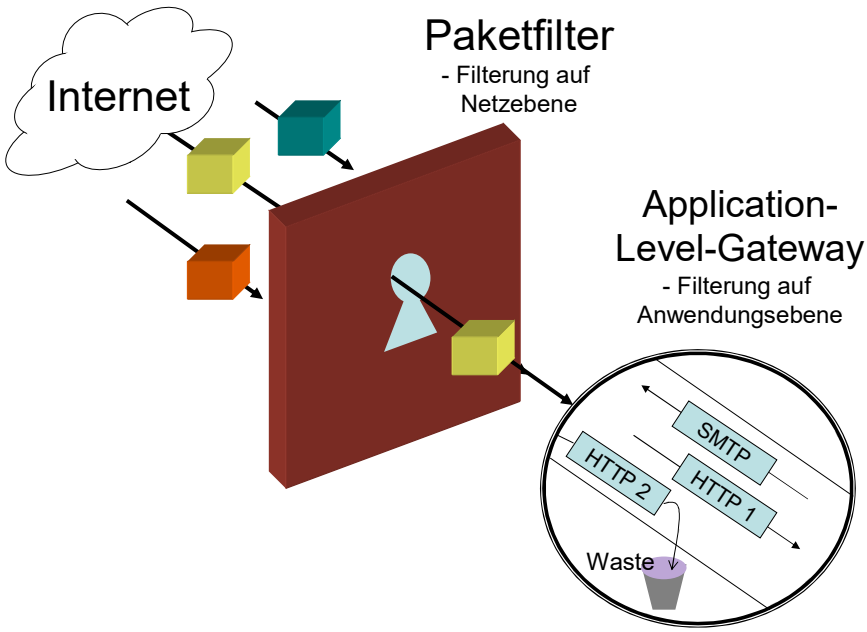


Application Level Gateway, Application Proxy

- Gateway stellt sich dem Client gegenüber als vollständige Applikation dar.
- Hierzu wird für jede Applikation ein dedizierter Applikations-Proxy benötigt.
- Bsp: Web-Applikation-Firewall (WebShield)



- Der Applikations-Proxy überwacht die übertragenen Daten auf Applikationsebene.
  - Z. B. wird hier verifiziert, dass die auf der Anwendungsschicht ausgetauschten Pakete den entsprechenden RFC Spezifikationen entsprechen.
- Am Client ist keinerlei Änderung erforderlich.
- Application Level Gateways können verschiedene Verfahren zur Authentisierung des Benutzers unterstützen
  - und ermöglichen somit ein nutzerspezifisches Logging
- Application Level Gateways bieten ein hohes Sicherheitsniveau
  - vollständige Abschirmung von Client und Server
  - bei einem erfolgreichen Angriff wird höchstens der Proxy-Server kompromittiert
  - lediglich Schwachstellen in der Applikation, die nicht im Datenstrom vom Application Level Gateway kontrolliert werden, können ausgenutzt/weitergeleitet werden



Zusammenfassung Firewalltypen

- Paketfilter (statisch):
  - Kontrolle auf der Netzwerkschicht
  - nur gerätebezogene Kontrolle
  - indirekte dienstbezogene Kontrolle über Portnummern
  - kein aussagekräftiges Audit
  - Vorteil: schnell und für die Dienste transparent
- Dynamischer Paketfilter (Stateful inspection)
  - zustandsbasierte Paketfilterung
  - dynamisches Regelwerk
  - im Vergleich zu statischen Paketfiltern deutlich verbesserte Sicherheit
- Circuit Level Gateways
  - Hoher Grad an Entkopplung
  - Geringer Grad an Datenflusskontrolle (wie statischer Paketfilter)
- Application Level Gateway, Application Proxy
  - Kontrolle auf der Anwendungsebene
  - Geräte- und benutzerbezogene Kontrolle möglich
  - Starke Authentisierung möglich
  - Aussagekräftiges Logging möglich
  - Nachteil: Für jeden Dienst ist ein Proxy notwendig

Personal Firewalls

- Wird auf dem Endsystem (typischerweise Client-PCs) installiert
- Kontrolliert den Netzverkehr des Endsystems
- Rein softwarebasiert, daher erhöhte Fehlerwahrscheinlichkeit
- Applikationsspezifische Filterregeln einfach realisierbar
- I.d.R. erfolgt Nutzung durch Nicht-Sicherheitsexperten (Normaluser). Daher automatische oder automatisierte Konfiguration
- „Must be“ für PCs am Internet

Proxy Server:

- Caching und Content Filtering stehen im Vordergrund

Proxy Firewall

- Inspektion der Daten auf Applikationsebene (in Kombination mit stateful inspection) steht im Vordergrund

Linux Firewalls (Open Source) vs. kommerzielle Firewall-Produkte

- Linux Firewalls
  - Open Source
    - » Unter ständiger „Überwachung“ der Internet-Community
    - » Sicherheitslücken, Patches
  - Kernel 2.4/2.6 bieten zustandsbasierte Filterung (iptables, netfilter)
  - Konfiguration technisch komplizierter als bei kommerziellen Produkten
  - Anschaffung preisgünstig, jedoch hohe Administrationskosten
  - Insgesamt i.d.R. höherer TCO (total cost of ownership) als bei vergleichbaren kommerziellen Produkten
  - Kommerzielle Produkte häufig Kombination mit anderen Sicherheitsfunktionen (Intrusion Detection/Prevention, Virenscanning, Content-Scanning, …)
    Bsp. Cisco Adaptive Security Appliance
- Firmen: Symantec, Cisco, Checkpoint, Juniper,…

Ausführungsformen kommerzieller Firewalls: Software Lösung vs. Appliance

- Appliance = Komplettprodukt inkl. Spezialhardware
- Vorteile von Softwarelösungen:
  - Hardware skalierbar, flexibel einsetzbar
  - Anbieter leicht wechselbar
  - Modularer als Appliance, verschiedene Anbieter kombinierbar FW, Content-Filter, Virenschutz („Best of breed“)
  - Teilweise Open Source
- Vorteile Appliance:
  - HW/SW aus einer Hand: weniger Schnittstellenprobleme
  - Minimaler, gehärteter Softwarekern:
    - » geringere Wahrscheinlichkeit von Schwachstellen
    - » neue OS-Schwachstellen betreffen nicht Appliance
    - » Geringerer Betriebsaufwand: Patchmanagement
  - Geringerer Installationsaufwand: lediglich Anwendungskonfiguration erforderlich

2 Firewall Architekturen

- Bislang wurden nur einzelne Firewall-Komponenten betrachtet
- Aus diesen Bausteinen werden jetzt Systeme am Netzübergang gebaut

Ausgangsproblematik:

- Zwei Netze unterschiedlicher Vertrauenswürdigkeit sollen verbunden werden.

Fail-Safe Konzepte

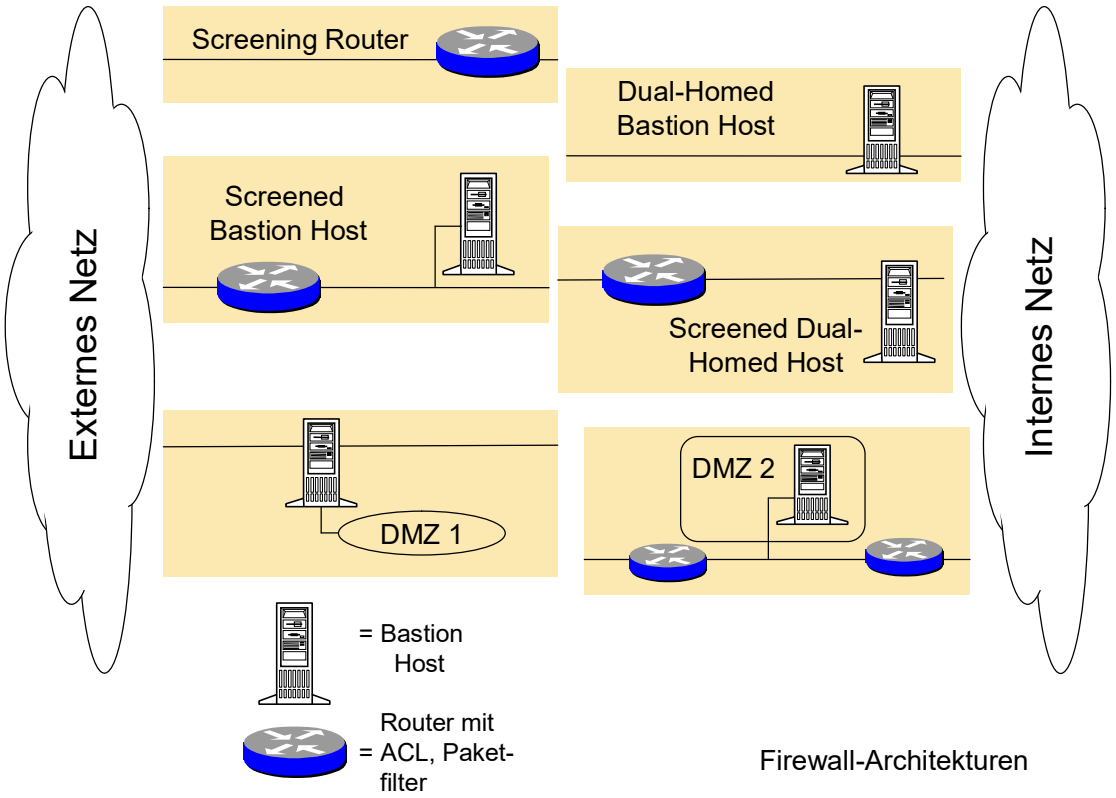
- Ziele
  - Sicherheit gegenüber einzelnen Fehlfunktionen einzelner Komponenten
  - Erhöhung der Widerstandsfähigkeit des Firewall-Systems
- Einsatz mehrstufiger Firewall-Architekturen (Reihenschaltung)
  - Beim Ausfall eines Elements ist das interne Netz weiterhin durch die verbleibenden Elemente geschützt
- Einsatz von Komponenten unterschiedlicher Hersteller
  - Redundanz gegenüber Schwachstellen in einem der eingesetzten Produkte

Screening Router

- Router mit Access Control List (ACL) arbeitet als Paketfilter
- Gesamte Sicherheit der Netzkopplung hängt am Router
- Transparente Dienstnutzung
- Gerätebezogene Datenflusskontrolle (ACL)
- Nur eingeschränktes Logging, höchstens gerätebezogen
- Lediglich Identifizierung von Paketen auf Basis von IP-Adr. und Portnummer

Dual Homed Bastion Host

- Gateway mit zwei Netzwerkkarten
- Gesamter Netzverkehr muss über das Gateway passieren (Bottleneck)
- Gateway kann sowohl als Paketfilter arbeiten als auch eine applikationsbezogene Filterung von Diensten (Proxies) ermöglichen
- Gesamte Sicherheit hängt am Gateway (einstufige Architektur)
- Gateway ist als Rechner Angriffen ausgesetzt und daher besonders zu schützen.



Firewall-Architekturen

Screened Bastion Host

- Gesamte Sicherheit hängt am Router
- Bastion Host ist durch den Router gegenüber Angriffen aus dem externen Netz geschützt
- Trennung von Paketfilterung und Proxy-Diensten
- Netzverkehr kann wahlweise
  - durch den Router gefiltert werden
  - vom Router direkt ins interne Netz weitergeleitet werden
  - vom Router zum Bastion Host weitergeleitet werden
- Bastion Host gehört streng genommen bereits zum internen Netz, lediglich logisch zur Firewall

Screened Dual Homed Host

- Kombination von Dual Homed Bastion Host und Screened Bastion Host
- Zweistufige Architektur

Demilitarisierte Zone (Demilitarized Zone, DMZ), Screened Subnet

Subnetz, das sowohl zum externen Netz als auch zum internen Netz hin geschützt ist.

DMZ1: Bastion Host / Screening Router mit drei Netzinterfaces

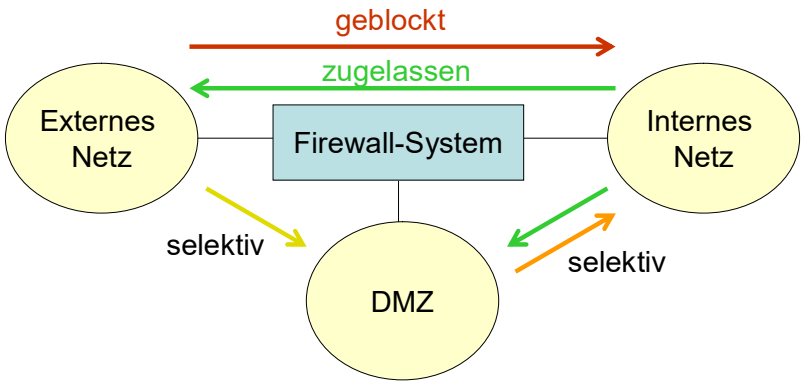
- Bastion Host regelt welcher Verkehr wohin fließen darf
- Einstufige Architektur

DMZ2: Interner/externer Paketfilter mit Bastion Host

- Bastion Host und andere Systeme in der DMZ sind durch Paketfilter gegenüber Angriffen aus dem externen und dem internen Netz geschützt.
- Sicherheit hängt an den beiden Routern
- Trennung von Paketfilterung (Router) und Proxy-Diensten (Bastion Host)
- Interne Netz ist geschützt, selbst wenn Bastion Host kompromittiert wurde

Typische DMZ Konfiguration

- Interne System dürfen Internet-Ressourcen und Ressourcen der DMZ nutzen
- Zugriffe aus dem Internet auf das interne Netz werden geblockt
- Zugriffe aus dem Internet auf Server in der DMZ werden selektiv zugelassen (http, https, smtp, dns, …)
- Zugriffe aus der DMZ auf das interne Netz werden (weitgehend) geblockt
  - ggf. DB-Anbindung aus DMZ nach intern für Anwendungen mit Transaktionen erforderlich (Shop, Homebanking)



Grundsätze für Firewall-Architekturen:

- Je höher die Differenz der Vertrauenswürdigkeit zwischen den zu koppelnden Netzen, desto mehr Stufen sollte das Firewall-System aufweisen.
- Serversysteme in internen Netzen, sollten nicht direkt aus externen Netzen erreichbar sein.
- Serversysteme, die aus externen Netzen erreicht werden müssen, sollten in speziellen Teilnetzen (DMZ) betrieben werden.
- Systeme in der DMZ:
  - kein Export kritischer interner Daten in die DMZ
  - kein Login auf internen Systemen aus der DMZ heraus
  - Logging z. B. über serielle Schnittstelle auf einen separaten Logging Server ohne Netzanbindung

Zoning: Aufteilung des internen Netzes in verschiedene Sicherheitszonen

- DMZ zum Betrieb aus dem Internet erreichbarer Systeme
- Produktionnetz mit Produktionssteueranlagen (SCADA-Systeme)
- Ein oder mehrere Servernetze zum Betrieb von Anwendungs-/Datei-/DB-Servern
- Ein oder mehrere Netze mit Arbeitsplatz-PCs
- Ggf. separates Netz für WirelessLAN Zugänge für mobile Geräte
- Gäste WLAN / Gastnetz

Internetübergänge sicherheitsrelevanter Anwendungen (z.B. Homebanking) haben häufig mehrere separate DMZs

- DMZ 1: Virenscanning / Intrusion Detection
- DMZ 2: Content-Filtering / WebShield
- DMZ 3: Webserver

Policy-based Firewalling

- Regelwerke gemäß benötigter Kommunikationsbeziehungen
- Automatische Konfiguration von Netzübergangskomponenten (Paketfilter, Bastion Host)
- Es gibt Produkte mit folgender Funktionalität:
  - Input: Benötigte Kommunikationsbeziehungen, Netzstrukturplan
  - Output: Regelwerke für sämtliche Netzübergangskomponenten

3 Firewall Organisation

- Sicherheitskonzept zur Firewall / Netzsicherheitskonzept
  - Netztopologie inkl. Kontrollkomponenten
  - Auflistung der Kommunikationsbeziehungen inkl. zugehöriger Kontrollen

- Betriebshandbuch: Beschreibung folgender Prozesse:
  - Freischalten neuer Dienste/Kommunikationsbeziehungen am Netzübergang
  - Sperren bestehender Dienste/Kommunikationsbeziehungen
  - Regelmäßig Überprüfung, ob alle Freischaltungen notwendig sind
  - Regelmäßige Auswertung von Protokolldateien