Lecture topics

- Firewalls
- Security model with firewalls
- Intrusion detection systems
- Intrusion prevention systems
- How to prevent and detect attacks

What is a firewall

- Divides network into two (or more) parts with different security policy
  - internal network ⇔ Internet
  - engineering ⇔ accounting: the other network must not be less secure that the other one. They just have different security policies or different assets to protect.
  - internal network ⇔ public servers ⇔ Internet
  - building automation ⇔ VoIP ⇔ surveillance system
- Enforces security policy
  - allowed traffic
  - prohibited traffic

Refer to IPsec security policy database (SPD): traffic is bypassed, discarded, or bypassed as protected.

- May have additional roles, such as VPN endpoint

Firewall types

Packet-filtering makes decision based only packet fields

- router ACL (access control list)
- TCP implicit state: for example to disallow incoming connections, firewall will drop any packet that has SYN flag set but no ACK and allows any packet with SYN+ACK.
- difficult with UDP, also some other TCP-based protocols such as FTP in active mode, where server establishes connection to client.

Stateful keeps track on connections

- maintains connection state
  - single point of failure
  - has to have some timeout mechanism as the state space is limited. Some attacks may exhaust state space.
  ⇒ random disconnections
• possible to accept related connections: for some protocols this needs application gateway.

**Application gateway** interpret connection on application level

• checks if application traffic is valid
• protects from simple port changes
• may provide payload inspection to detect malicious payload
• proxy servers
  – call-out
  – in-line (transparent)

**Address-translation** between internal numbering and external addresses

• using NAPT provides same as prohibiting incoming TCP
• internal topology can be hidden

**Host-based** or software firewalls add on application security

• completes application security and access control
• possibly user- and application-level control

**Hybrid** use combination of different types for performance

• check start of connection with application gateway, switch to stateful filtering ⇒ better performance as bulk of traffic is handled by fast path.

**Firewall topologies**

![Diagram of Firewall Topologies](image)

**Building firewall rules**

• Defining default policy
  
  – “everything not prohibited is allowed”
  * “router” ACL
  * enumerate vulnerable services and protect them
  – “everything not allowed is prohibited”
  * enumerate needed and safe services and allow them
  – both policies need continuous updating

• There should be one rule for one packet
  
  – multiple overlapping rules
order of rules matters

– performance issues: hardware-based routers/firewalls can handle certain number of rules without significant performance penalty. For software-based firewalls order of rules does matter.

• Possibility to oversight
• High-level languages not solution

Deploying multiple firewalls

• Helps to limit the impact of attack
• Protection by diversity
  – on other hand, multiple systems to update
• Designing rules even more complicated

What firewall protects and what not

• Protects
  – from known, vulnerable protocols
  – static network configuration
• Does not protect for / from
  – executable/active content
  – malicious insider
  – loopholes: modems, WLAN, mobile networks
  – carry-in attacks such as notebooks, mass storage
  – new attacks
  – most DoS attacks
• May result “hard perimeter, mellow inside”
  – failure to update internal systems
  – selecting insecure protocols and applications

Security in organisation
How secure are firewalls

- Common Vulnerabilities and Exposures: 110 matches on “firewall”
  
  **Check Point FireWall-1** 34 entries  
  **Cisco** 13 entries  
  **Juniper** 1 entry  
  **Linux** 6  
  **Symantec** 17  
  **WatchGuard** 11 entries

- More features (VPN, virus checks, QoS protection)  
  ⇒ more code  
  ⇒ more bugs  
  ⇒ more vulnerabilities

Intrusion Detection Systems

- How to make sure that firewall is not leaking  
- How to detect internal attacks  
- IDS is designed to  
  - detect,  
  - identify, and  
  - report malicious activity  
- IDS can be located different places  
  - application  
  - host  
  - network

Application and host IDS

- Application instrumented to identify abnormal actions  
  - high level of abstraction  
  - user actions monitored  
  - policy violations  
  - application log analysis  
  - access to encrypted data  
  - may not protect application flaws  
- Host instrumented  
  - reference monitor  
  - actions by user and application  
  - host log analysis  
- Log analysis best on separate host  
  - provides after-the-fact analysis  
  - vulnerable to network attacks
Network IDS

- Monitors traffic
  - best done with signal splitters

- Large volume of data
  - low level of abstraction
  - encrypted traffic problematic

- Mostly misuse detection
  - recorded patterns of misuse (signatures)
  - frequent updates (like virus scanners)

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 22
(msg:"EXPLOIT ssh CRC32 overflow /bin/sh";
flow:to_server,established;
content:"/bin/sh"; )
```

- Anomaly detection
  - detecting differences to normal
    * threshold detection
    * statistical profile
    * rule-based detection
  - learning system

- Large number of alerts
  - 3700 alerts from corporate network per day
  - 48 should be studied in detail
  - 2 warrant an action

IDS in large network

- One should monitor every link
  ⇒ very expensive

- Select important links
  - full census on those

- Do random sampling on other links
  - if one samples every 512th packet
    ⇒ not a big increase in traffic
  - large problems notified immediately

Honeypots

- A false system similar to production system
  - all access illegal
    ⇒ any accessing is potential intruder

- Used as part of IDS
  - a connection results monitoring

- How to keep attacker from telling difference from real system
  - should be not too weak
  - should have “real” data and traffic
  - if virtual host, should not be visible
IDS reaction too slow

- IDS identifies attack
  - analysis may not be real-time
  - corrective actions may take time
- Epidemic security problem may be instant
- System can be scanned, attacked, and compromised in a minute or less
  \( \Rightarrow \) Need for automation

Intrusion Prevention Systems (IPS)

- IDS with automatic response
- Suffers from large number of false alerts
- A firewall with automatic ACL update
- Virus scanners are host-based IPS
- Still at early stages
  - does not stop vendors from marketing...

Traffic traceback

- Problem: where incoming attack traffic originates
- Source IP cannot be trusted
  - sender can put it to any address
  - ingress filtering not deployed universally
- Should not need additional hardware or load on routers
- Scalability problems, few proposals

Security in Ad-hoc networks

- Ad-hoc networks interesting topic
  - self-building topology
  - extending network coverage
- Must rely on other hosts
  - no central authority, block lists
  - no trusted core network
  - routing done by devices
- Public key-based per-packet authentication too heavy
  - modern PC throughput few ten kbit/s
- How to communicate trustfulness?
Challenges in All-IP world

- Large number of non-technical users
  - the “---:--” generation
  - rightful ignorance: I want to watch movies — fixing security problems does not match to my idea of relaxing.
- Service provider responsibility
- Multi-vendor environment

Summary

- Firewall and IDS are good tools
- Must know their limitations
- Future challenges
  - accurate detection of malicious activity
  - security in ubiquitous computing
  - trust in autonomous systems

Easter holiday 2005-03-29, no lecture

References


