Internet
solutions

Malware, Intrusion Detection and E-mail Security

Nixu Oy

PL 21

(Mäkelänkatu 91)

00601 Helsinki, Finland

tel. +358 9 478 1011

fax. +358 9 478 1030

info@nixu.fi

http://www.nixu.fi

nixu

Agenda

- Viruses and other malware
- Intrusion Detection Systems
- Securing e-mail

Malware

- Software with a malign purpose
  - Viruses, troijan horses, worms etc.
- Usually created on purpose
- Can:
  - prevent correct use of resources (DoS) or cause general malfunctions
  - destroy information
  - transmit information to unauthorized parties, also to random addresses
  - enable unauthorized parties to have a complete control over a computer

Types of malware

- Trojan horses
  - programs that promise something good, but instead or in addition do something nasty when you run them
  - examples: root kits, games with backdoors, etc.
  - do not spread automatically
  - can open existing security vulnerabilities or create new ones
  - some software allows complete control of the target host

… Types of malware

- Viruses
  - self-replicating softeware
  - attach themselves to other executable content
    - > programs
    - > boot sectors on disks
    - > documents with macro instructions
  - currently the biggest problem are macro viruses in various Windows software
  - MS Office macros, Visual Basic
- Worms
  - Network aware viruses that propagate as independent programs and usually use security vulnerabilities to enter different host computers

… Types of malware

- Hostile Java applets and ActiveX components
  - user may not even notice he is running an active component on his computer
  - component certification doesn't mean the component cannot be a Trojan, it is only intended to identify the source of the software component
  - getting an user to run a trojan program is an easy way to get a foothold inside a network
  - Java's sandbox is in theory fairly secure, but vulnerabilities have been found
  - ActiveX does not limit the component in any way
- "Remote administration" programs
  - examples: NetBus, Back Orifice
  - can be packaged into trojan horses

How does a Virus Spread?

- For example a simple MS-DOS virus:
  - moves part of a program's binary code from the beginning of the file to the end and places itself in the beginning
  - when the program is run the virus activates and places its code into the computer's memory and hooks to the operating system so that the code is periodically activated
  - then replaces the original program's code to the correct place in memeory and allows the program to be executed
  - when the OS activates the virus in the memory, it infects other files and possibly performs some additional tasks

How do Macro Viruses Spread?

- Macros are small application specific programs stored in the application's data files
  - the structured environment and high level services make life easier for the viruses
- The viral macro is stored among other macros and is configured to automatically activate when the document is opened by the application
  - after activation the macro can easily copy itself to other documents
  - the macro can also use other available services and for example send random documents using e-mail
- The macro viruses can also use security weaknesses to for example force themselves be activated when an e-mail message is just read into a specific reading program

How to defeat viruses

- Avoid environments that support viruses (e.g. Microsoft Office tools)
- Use a virus scanner that knows the signatures of different viruses
  - the virus signature database needs to be updated frequently
  - virus scanning program manufacturers currently share new viruses efficiently and focus on keeping the scanning programs up to date
  - heuristic scanning that would recognize “bad intentions” of a program has been proposed frequently, but it does not yet work
  - the viruss scanner can remove the virus from the host file or destroy the file
  - the scanning can be done for every file when it is opened
  - the scanning can also be done to file servers or at firewalls

What is Intrusion Detection (ID)?

- E. Amoroso: Intrusion Detection is the process of identifying and responding to malicious activity targeted to computing and network resources.
- Houses may have surveillance cameras and burglar alarms; information systems may have intrusion detection
- Another analogy: network management systems (SNMP)
- Categories:
  - attack detection
  - intrusion detection
  - misuse detection

Intrusion Detection Characteristics

- Preferably automated identification of problems
- Monitors a whole system or just some part of it
- May be done before, during or after an intrusion
- May be stealth or openly advertised
- Makes an alarm if suspicious activity is detected and produces reports on long term development
- An administrator (a person!) must process the alarms
- Some systems also prioritize alarms and/or perform automated response

Why detect intrusions?

- Knowledge of ID may scare intruders off (at least it keeps the honest people honest)
- Measured figures of actual attacks help establishing a budget for security administration
- You have a chance of reacting to the attack:
  - You may be able to stop the intrusion before anything catastrophic happens
  - You know what has happened so you can manage the damage
  - You can try to stop it from happening again
- If you are not going to have competent and timely response to the alarms, you probably should not bother with the rest of intrusion detection either!

Intrusion Detection Methods

- Audit trail analysis
  - host based, may combine logs from many different sources
  - usually off-line, after-the-fact
- Network traffic analysis
  - on-line
  - parses the traffic to detect prohibited content (certain packet types, URLs...)
- Signatures of abnormal behavior
  - vs. virus scans
  - can detect only previously known attacks

… Intrusion Detection Methods

- Profiles of normal behavior
  - can also detect new attacks, assuming they differ from “normal” behaviour
  - difficult to define what is “normal”
- Heuristic analysis
  - artificial intelligence, neural networks, self organizing maps...
  - these methods do not generally have a good reputation, they generate too many false positives, however they might develop to something useful
- An actual system may use any combination of these methods

Network based Intrusion Detection

- Detecting attack signatures
- Traffic analysis
- Why network based ID?
  - Most attacks happen over the network
  - Fairly easy to do
  - Does not affect the speed of the network or the load of the hosts being monitored

Methodology

- Two-part architecture: one or several sensors and one analysis station
  - the more complex you network is, the more sensors you should have
  - a central analysis point makes correlation much easier
- If using only one sensor, place it at or near the firewall
  - outside firewall: catch information about outside attacks and inside attacks directed at outsiders, but miss the attacks in the inside network
  - inside firewall: catch also internal attacks, but miss attacks to the DMZ and attacks filtered by the firewall
  - at the firewall: can catch both internal and external events, but this may cause a performance degradation
- For best results, feed also your security policy rules to the IDS

Detecting attack signatures

- Most known attacks are based on a certain pattern of packets, e.g.
  - land: source address and port == destination address and port
  - echo-chargen oscillation: spoofed packet(s) from echo port to chargen port or vice versa
  - smurf: (spoofed) echo request sent to broadcast address
  - port scans: relatively large amount of similar packets from one source to different ports or different hosts within a short period of time
- Certain strings can be matched
  - e.g. /etc/passwd
- The problem with signatures is you cannot detect anything that doesn't match your signature definitions

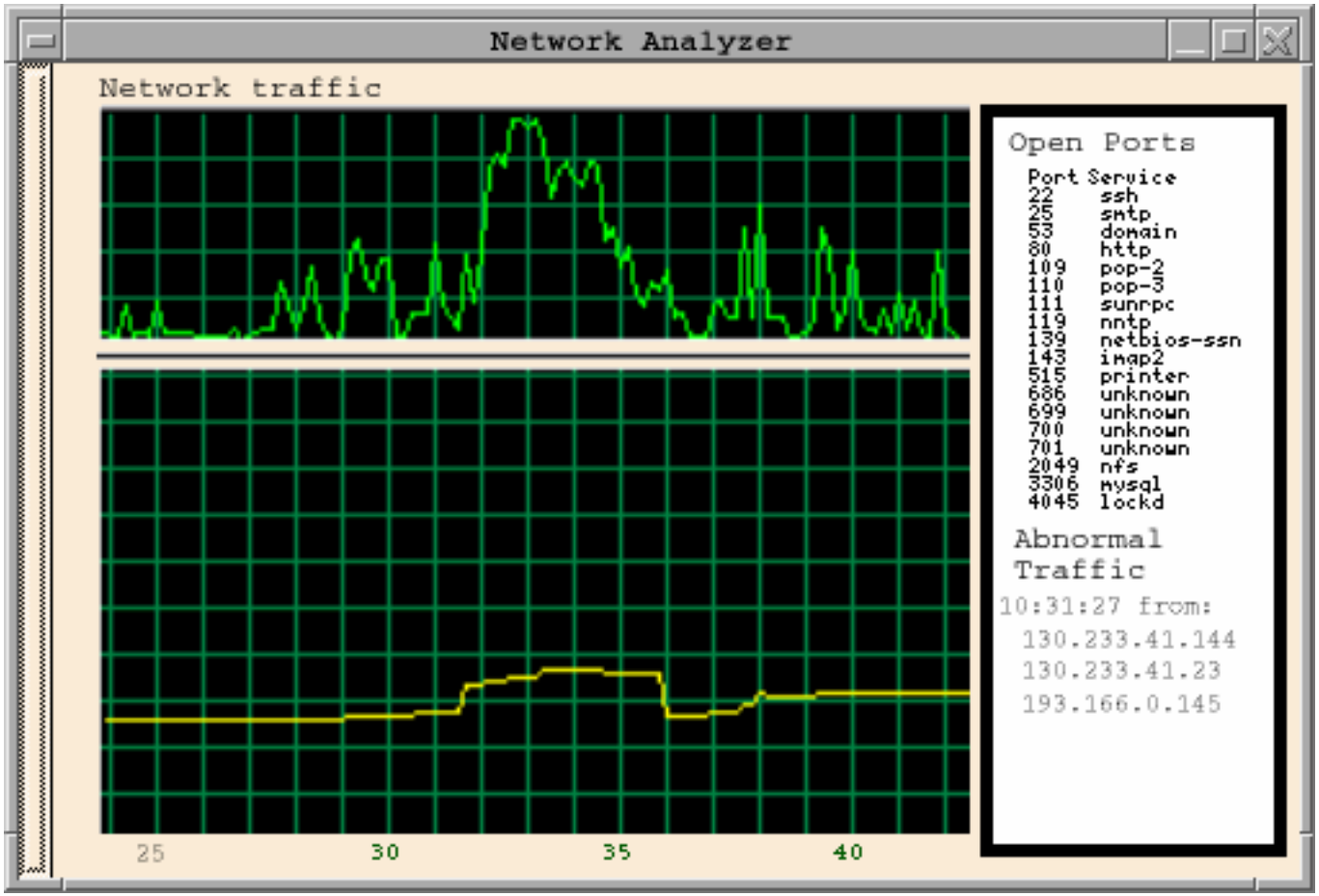
IDS vs. anti-virus software

- Virus detection software can detect many known Trojan horses
  - NetBus, Back Orifice
- Most companies have anti-virus software; using IDS is still rare
- The anti-virus software can detect thousands and thousands of virus signatures; ID systems only have signatures for a few hundreds of attacks
- Most anti-virus vendors have daily updates available from the Web; ID vendors issue updates a couple of times a year
- The virus detection community shares signature information much more effectively than the ID community

Traffic analysis

- Statistical analysis with tolerance limits is a good starting point for detecting new attacks
- Abnormal behavior is always a reason to investigate
  - unusual amount of traffic
  - traffic between hosts that do not normally talk to each other
- You do not always have to see the traffic contents to detect that something weird is going on
  - encrypted connections
  - unknown protocols
  - traffic volume
- To actually recognize an attack, you usually need more information

Traffic analysis (cont.)



IDS vs. Network monitoring

- Similarities: monitoring the network, raising an alarm if problems are found, usually needs human intervention to fix the problem
- In some cases, a network management tool may also work as an ID tool
  - can detect Denial of Service
  - can detect error conditions that may be a result of an attack
  - can detect anomalies in network traffic and load
    - > e.g. a switch that changes to hub mode can indicate an eavesdropping attack
- If you are using a network monitoring system, use it to get an idea of what is normal in your network

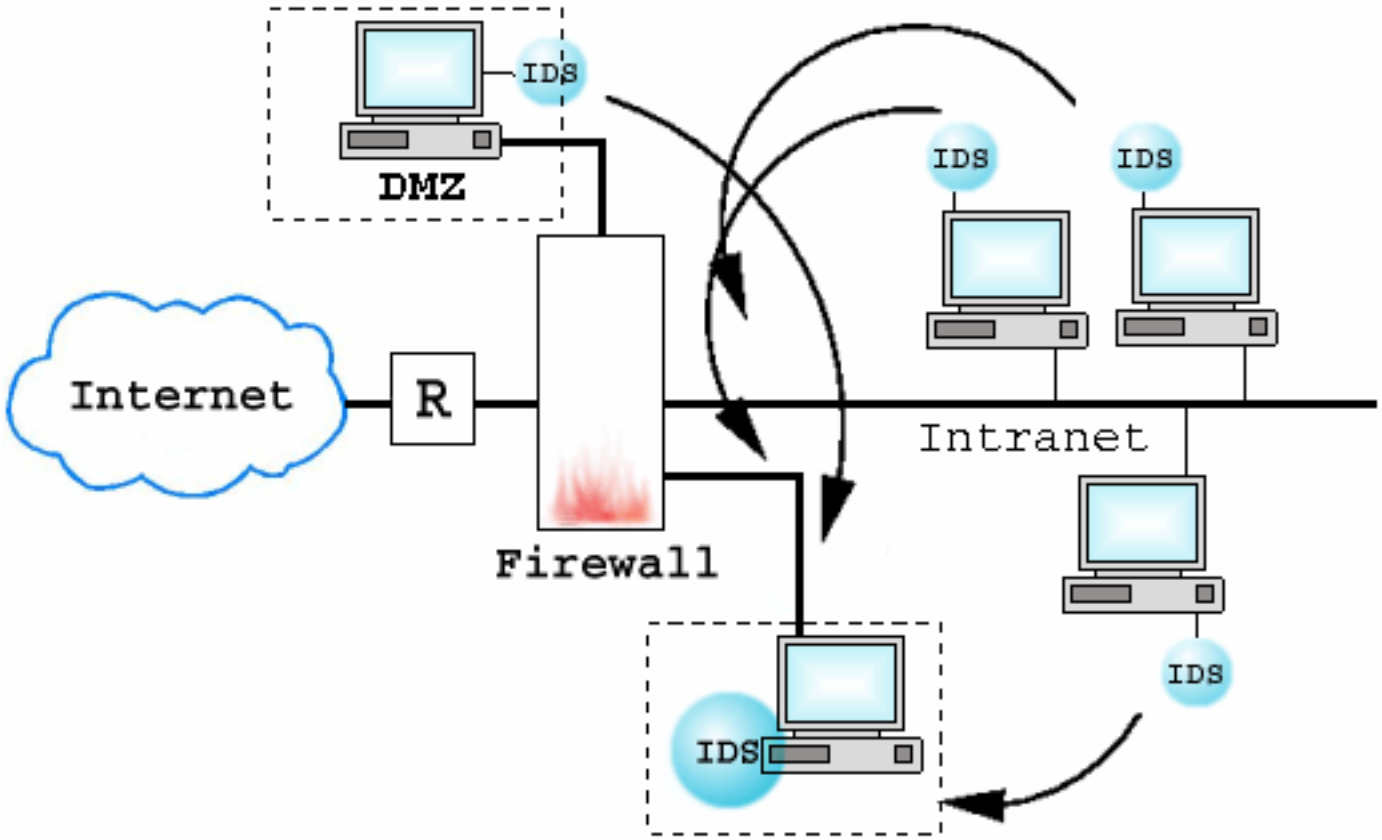
Host based Intrusion Detection

  - Checksums, rootkits
  - Heuristic analysis
- Why host based ID?
  - Network load often exceeds the processing capability of network detectors and analysis
  - Encrypted network traffic cannot be analysed
  - Switches may not allow the IDS to see all traffic
  - The IDS may not understand all protocols (a common problem!)
    - The host may interpret the traffic in a different way than the IDS
  - Not all attacks happen over the network
    - Insiders
    - Modems and other ways to bypass the network
  - Consequence: network based ID may miss some attacks

Methodology

- Host systems monitor their internal events, network connections and file system status and create logs
- All critical systems (and preferably others as well) send their logs to a central location for
  - more secure storage
  - parsing, analysis and correlation
- The host may also do local processing and alerting (no single point of failure)

Multi-IDS setup



Checksums and cryptographic hashes

- Basic idea: calculate a set of digits from a file
- If the file changes, calculating the digits again will result in different digits and the change can be detected
  - the bad news is the attack has already happened: this is not prevention
  - the first checksum/hash must be calculated from a clean system
- Checksums can be defeated by carefully crafting the change
- Cryptographic hashes are much harder to trick
- Calculating more than one type of hash (e.g. md5 and sha1) of the file is secure enough for most situations

Local Access Control Lists for Servers

- Some tools allow monitoring and filtering of incoming requests for network services
  - without modifications to existing server software or configurations
- They catch scans and attacks that come from the network
- Example: TCP Wrappers
  - ftp://coast.cs.purdue.edu/pub/tools/unix/tcp_wrappers/
- TCP Wrapper configuration:
  - log everything
  - deny everything (`/etc/hosts.deny ALL:ALL`)
  - allow only specific services and hosts that we trust (`/etc/hosts.allow`)
  - use paranoid mode (checks both forward and reverse DNS lookup)

System logs and log monitors

- Most systems collect error messages, warnings and other messages to some kind of system log
  - Unix: syslog
  - NT: System/Security Log
- Low tech ID: manual inspection of the logs
  - Unix: `/var/log/*`
  - NT: Administrative Tools –> Event Viewer –> Security
- There are also tools for monitoring the logs and raising alerts
  - Unix: swatch (free), CMDS (commercial), LogSurfer
  - NT: PageR Enterprise (commercial)
- Some operating systems have the possibility to collect more elaborate audit logs

Deception, honeypots and traps

- The attackers are using deception, why should not you too?
  - e.g. give out wrong version information etc. to make the attacker try the wrong exploits
- You could build custom traps
  - phony accounts with crackable passwords that generate alarms when used
  - phony watchdog processes or interesting looking files that generate alarms when read
  - out-of-band traps for trace back
  - even look-alike shells written in Perl exist
- Target:
  - Divert attacks to unimportant systems
  - Make the attacker waste time
- Warning: building traps and requires competence and the advice of a lawyer

Intrusion Detection Framework

- How secure are you?
- Protection + Detection + Response
- Do you have proper protection?
  - prevention is always better than cure
- Can the attacker learn what your IDS does and does not notice?
  - pushing vs. pulling the data from the sensors
- Can the attacker attack the ID system first to keep you from noticing when he goes after the protections?
- Can the attacker attack the response channel to keep you from responding to the detect?

Reporting

- Reporting helps the security team and the management (security budget)
- Report generation from the IDS should be as automatic and easy as possible
- Event Detection Reports
  - event-by-event or daily summary
  - for the analyst and the incident response team
  - low-level, detailed information about the detected attack
  - usually sent as an email to the relevant parties (should be encrypted)
- Summary Reports
  - monthly or daily
  - targeted at management
  - high-level information, “the big picture”

Different types of response

- Human initiated response
  - restoring the system, or whatever action is appropriate
  - pros: flexible, can handle false positives and new situations
  - cons: slow
- Automated response
  - dropping attacker network connections, closing accounts, active counter attacks etc.
  - pros: fast
  - cons: unable to make judgement calls (handling false positives)
- Coordinated human and automatic response
  - when done right, can make the best of both types of response

Things to consider

- The nature of the incident:
  - What assets have been affected by the incident?
  - Has this sort of incident happened before?
  - How did the incident happen?
  - Can we trust the information source?
  - Can the evidence be correlated with other information?
- The effect of response:
  - What if we modify or reduce target system functionality?
  - What if we initiate traps or trace back?
  - What if we do nothing?
  - Is the response legal?
  - Who should be involved?

Incident response: things to keep in mind

- Follow your organization's policies and procedures
  - maintain chain of command
  - have the procedure checklist ready and handy
- Contact appropriate incident response agency (e.g. CERT)
- Keep communication out-of-band
  - while investigating, limit the number of knowledgeable people to the minimum
- Document your actions
- Make copies of all files that can serve as evidence
  - the best is a raw copy of your hard disk (backup software may change access times)
  - store them securely off-line, document Chain of Custody

Handling false positives

- All ID systems get false positives (up to 90% of all detects)
- When ever you detect an attack, you should ask yourself:
  Is this a real attack, or is it a false positive (e.g.: a configuration problem or legitimate traffic that just looks a lot like an attack)?
- The IDS should allow you to easily view all the relevant data, so you can make the decision
- It always helps if you can correlate the attack with some out-of-band information, like calling the attacking site and asking what's going on

… Handling false positives

- If some rule in you ruleset generates a lot of false positives, maybe it's time to tune the rule a bit
  - to do that you must understand what the rule is all about
  - if your IDS does not allow tuning, you may want to change the product

Staffing

- No matter what your IDS software and hardware cost, the really expensive part is getting the people to take care of the response
  - having someone on-call 24 hours a day, 7 days a week requires 10 people absolute min-imun, more likely 20 people
- Incident analysis takes the mindset of a detective and the ability to work under a lot of pressure, in addition to a lot of technical skill
- If you outsource this, make sure your contract states clearly what you are getting, and be prepared to pay for it
- If you are selling this sort of service, beware of too heavy overbooking

Legal issues

- Lures and honeypots may be considered encouragement to crime
- Many IDSes can also be used for other purposes than ID
  - looking for other criminal activity by employees (e.g. drugs)
  - monitoring web browsing (e.g. pornography)
  - looking for criticism targeted at managers
- While you can do it, you probably shouldn't!
  - Targeting a single person's network traffic is analogous to wiretapping his telephone (you need a warrant)
  - Monitoring email content is illegal in Finland

… Legal issues

- If you do monitor network traffic, always give clear notice
- Talk to your lawyer before setting up traps or monitors for anything other than intrusions

Future trends

- Standards
  - CIDF: standardized data exchange between the different ID components
  - several commercial interoperability solutions
- Combining different approaches
  - both network and host based ID
  - combining ID with network management or antivirus products
- Better correlation
  - source IP, destination IP, longer time windows etc.
- Better automated response

Criticism of IDS

- ID is still not easy enough (and is likely to stay that way)
- Most ID systems do not use enough correlation to be efficient
- ID is by nature fail-open
- ID system may be used by crackers to create a diversion to camouflage the real attack

Where does IDS work

- In specific cases where the system behaviour is well know IDS can work perfectly
  - Between an e-commerce front-end and back-end
  - Limited purpose servers
  - When the system protected is clearly documented and the documentation matches reality
- In a general network environment running an IDS is more messy

Does it really work?

- A lot of it is still just hype
- Difficult to define what an intrusion is; how can you measure something you cannot define?
- There is no ideal Intrusion Detection System (IDS):
  - if you have 100% detection, you have a severe problem with false positives
  - if you have 0% false positives, you are likely to have ~0% detection as well
- If you understand the limitations, IDS is a good tool in your security toolkit

Securing E-mail Servers

- The basic SMTP protocol used for transmitting mail messages over the Internet contains no security features:
  - It is possible for anyone to read your e-mail, as it travels in public networks
  - It is generally easy to send e-mail messages with a forged sender address
  - PGP and S/MIME address these issues
- But there are other problems
- Server software has known problems
- Spam is nowadays quite a nuisance
  - There are some technical measures to reduce spam
- Protocols used to remotely read mail are somewhat insecure

Server software

- Sendmail is usually delivered as a default SMTP server on UNIX boxes
  - Very powerful configuration
  - Which is not human-readable
  - Not particularly efficient
  - BAD track record on security incidents
    - Starting with The Great Worm of 1988
    - SEEMS to have settled as of 13.1.2000
  - Basic design is somewhat insecure
  - Runs as root
    - MAJOR problem

Hardening sendmail

- If no local delivery (e.g. relay only), can be run as non-root
  - But still needs root access for some time to be able to listen on port 25
- Can be run chrooted
  - But if run as root, this does not offer complete protection
- Shell delivery can be restricted by using smrsh
  - Cheap way to avoid some problems

Replacing sendmail

- The main alternative for UNIX platforms would be qmail
  - Hardened architecture
  - Good track record
  - Less features
    - > E.g. spam-prevention worse than in sendmail
  - Not so well-known
  - Configuration is simple
  - Installation requires work
    - > 7 different user accounts
  - Things are done in Different Ways (compared to many others)

Server software

- Some POP and IMAP servers have had major security problems
  - Buffer overflows etc.
- These holes are widely known and exploited
  - So be careful with these

Anti-spam measures

- Sendmail is dominant MTA on UNIX systems
- Recent versions have good anti-spam features
  - No relaying by default
    - Nobody should be able to use your server to spam others
  - Sender domain must resolve from DNS
    - Sender addresses must be replyable
  - Possibility to have short-circuits based on domain or IP
    - Allowing mail which would be denied
    - Denying mail which would be allowed

Anti-spam measures (cont.)

- Even more strict measures can be taken
  - DNS reverse map for sender must exist
    - > This blocks some legitimate senders
- Black lists can easily be enabled
  - RBL (Realtime Blackhole List)
    - > Tight policy (to get here requires some effort)
  - DUL (Dial-ups)
    - > Medium-tight policy
    - > Helps to cut down spam sent from dial-up accounts
  - ORBS
    - > Loose policy

Remote mail reading

- Nowdays, mail is typically downloaded from server to workstation and processed there
  - POP, IMAP
  - Standard solution is to give users shell accounts on server and use same usernames and passwords for both shell work and mail reading
- These protocols send data unencrypted
  - Usernames, passwords and mail messages can be sniffed from the network
    - > Usernames and passwords can be used to gain access to server
    - > Mail messages can be read

POP / IMAP solutions

- Make special user accounts for mail reading only
  - No shell access, can not be used to gain access to server
  - Mail itself is still vulnerable
- Use authentication method which does not transmit password in cleartext
  - POP3: APOP authentication
  - Not all clients/servers implement this (not mandatory)

POP / IMAP solutions (cont.)

- Tunnel POP / IMAP sessions with something secure
  - This is security-wise best solution
  - SSH
  - SSL (with long enough keys!)
  - IPSec