Imperial College
London

Network and Web Security

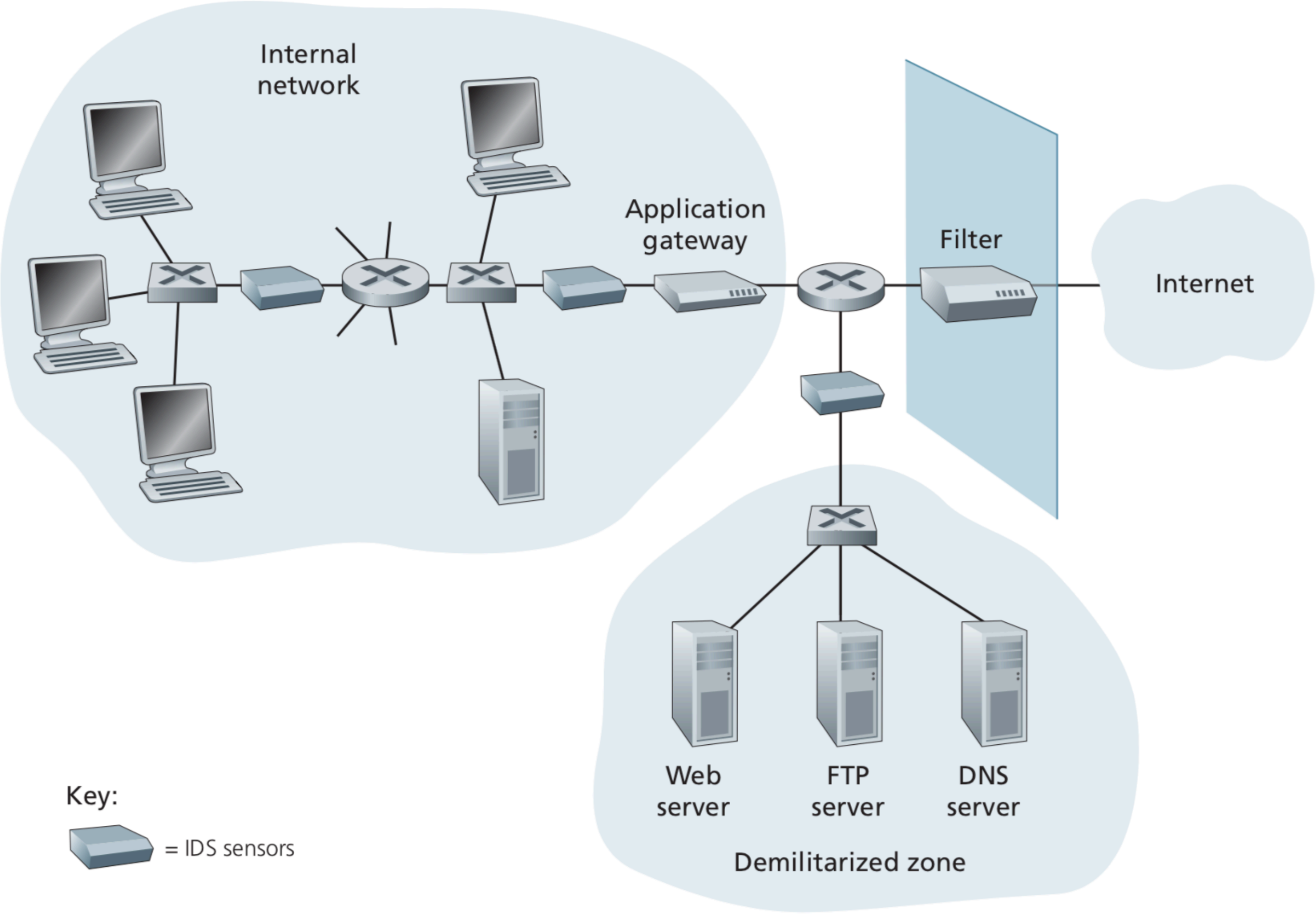
Network defenses

Dr Sergio Maffeis
Department of Computing
Course web page: https://331.cybersec.fun

Network defenses

Imperial College London

- Main firewall (or "filter") protects **all** internet traffic
  - Enforce protocol- or port-based access control
- Internal network is kept separate from Internet-facing services (DMZ)
- Application gateway(s) control connectivity of a specific application
  - (Authenticated) user-based access control
- Intrusion Detection Systems (IDSs) at different points of the network detect attacks

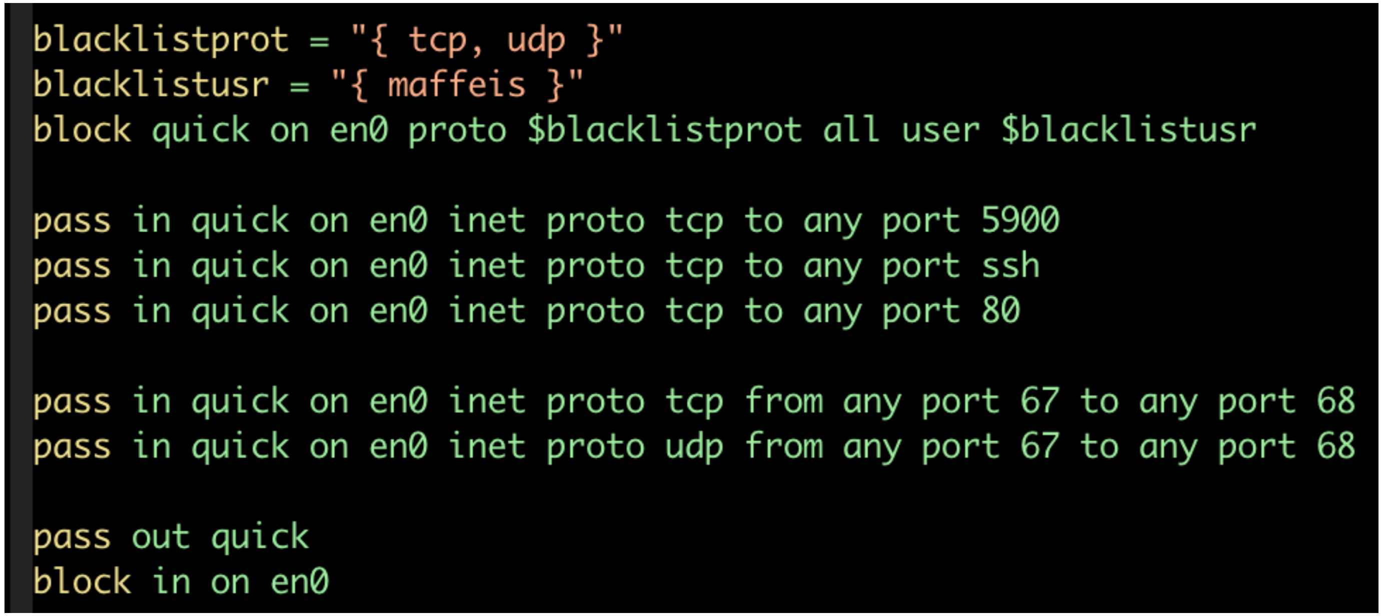


Firewalls

- Aim
  - Interpose on all inbound/outbound traffic
  - Enforce security policy
    - Allow/disallow certain *kind* of network traffic
    - Likely to also prevent attacks, but not main/only goal
- Main firewall tends to be centralised
  - Easier to maintain, enforce global policy
  - Possibly other dedicated firewalls on subnets
  - Most modern hosts have their local firewalls too
- Dedicated network appliances with purpose-built hardware
  - Cisco, CheckPoint, etc
- Kernel-level applications on general purpose hosts
  - `iptables` (Linux), `pf` (Open/FreeBSD, OSX), Windows firewall, etc
- Firewalls are valuable target for attackers
  - Privileged position on victim network or OS
  - Example: 0-day in Cisco firewall revealed by NSA leaks

Firewall policies

- Packet filters: make decision based on individual packet
  - Protocol header fields IP/TCP/UDP/ICMP, in particular source/destination addresses, ports
  - Network interface used
- Stateful filters: take different notions of state into account
  - Is packet part of an established connection?
  - Timeouts, bandwidth usage
  - Test/set variables in FW policy script
- Modern firewalls tend to support both kind of filtering
  - And some payload inspection as well
- `pf` example below of a paranoid policy
  - prevent tcp/udp for a specific user (block “my” traffic while pentesting via a proxy)
  - serve VNC, ssh, HTTP
  - allow DHCP
  - allow other outbound
  - block other inbound



```
blacklistprot = "{ tcp, udp }"
blacklistusr = "{ maffeis }"
block quick on en0 proto $blacklistprot all user $blacklistusr

pass in quick on en0 inet proto tcp to any port 5900
pass in quick on en0 inet proto tcp to any port ssh
pass in quick on en0 inet proto tcp to any port 80

pass in quick on en0 inet proto tcp from any port 67 to any port 68
pass in quick on en0 inet proto udp from any port 67 to any port 68

pass out quick
block in on en0
```

Intrusion Detection Systems

- Beyond firewalls: *deep packet inspection*
  - Make decisions based also on the payload, not only headers
  - Raise an alert (IDS)
  - Drop packet (IPS = Intrusion Prevention System)
- Aim to detect/prevent attacks
  - Active information gathering: host/port/vuln scans, port sweeps
  - DDoS
  - Worms
  - Application layer attacks
- Free/open source frameworks: Snort, Zeek, Suricata
- Main conceptual approaches
  - Signature based
  - Anomaly detection based
  - Specification based
    - Positive and negative logical rules to restrict traffic
      - Automated reasoning, Bayesian inference
    - Hard to define rules, avoid conflicts, inconsistency

Signature based IDS

- Aka *misuse based IDS*
- Database of *signatures*
  - Sets of rules on individual or groups of packets that aim to detect an attack
  - Sensitivity/specificity trade off
    - Generalising rules to catch attack variants increases false positives
  - Mostly human-expert-generated
  - ModSecurity example

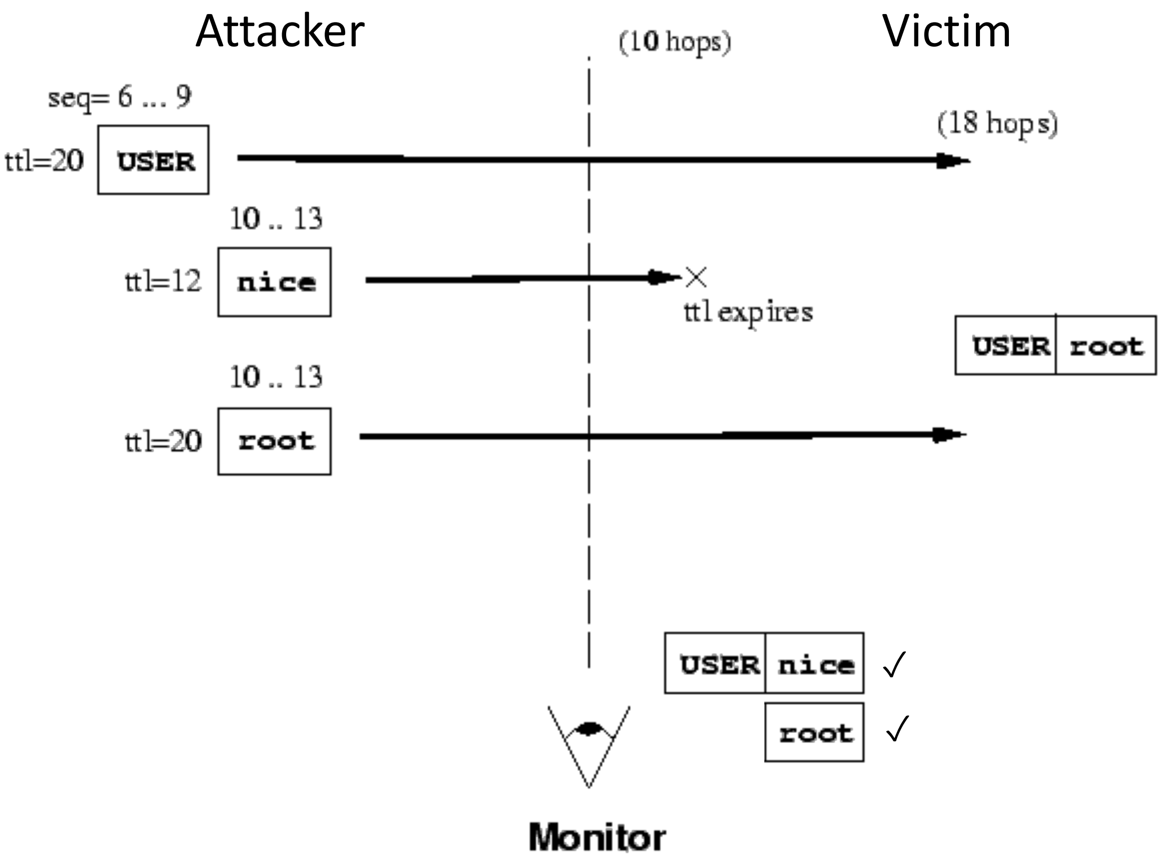
```
SecRule ARGS|REQUEST_HEADERS “@rx <script>”
id:101,msg:‘XSS Attack’,severity:ERROR,deny,status:404
```

- Good at detecting familiar attacks with low false positives
- Not future-proof
  - Unable to detect unanticipated attacks
- Signatures are easy to bypass
  - Same situation as antivirus and polymorphic malware
- Focus on content rather than intent
  - Better at stopping automated attacks (Angler, Neutrino, RIG) than manual/targeted ones
- Performance
  - Each rule may be simple
  - Applying each rule to each packet may become prohibitively expensive
  - False negatives under bandwidth stress

IDS evasion

Example of attack against IDS using IP fragmentation:

1. Fragment a suspicious IP packet in 2
2. Traceroute to determine distance to IDS and target
3. Send frag 1 to reach target
4. Send innocuous replacement of frag 2 so that it's seen by the IDS but expires before reaching the target
5. IDS decides that communication is safe
6. Send malicious frag 2 so that it reaches the target
7. IDS does not interpret message from (6) as related to the one in (3)



(*A System for Detecting Network Intruders in Real-Time*, V. Paxson)

Anomaly detection based IDS

- Network IDS
  - Learn a model of benign traffic
    - Data is a multivariate time series, with heterogeneous features: categorical (TCP/UDP), continuous (size) …
  - Report statistical anomalies
- Pros and Cons
  - Can detect previously unseen attacks, for which there is no signature
  - Tend to suffer from many false positives
    - Sensitivity/specificity trade off for model parameters, thresholds
- Type of anomalies
  - Point: one sample is anomalous wrt the others
  - Contextual: one sample exhibits behaviour which is anomalous in a specific context, but not in others
    - Example: high bandwidth usage (behavioural feature) during nighttime (contextual feature)
  - Collective: a set of samples is anomalous wrt all the samples available
    - Example: TCP connect scan to same port of many different hosts (port sweep)
- Models used for network security
  - Classification-based: Bayesian networks, Neural networks, SMVs, Random forests
  - Statistical: Non-parametric (Histograms, PCA), parametric (Regression)
  - Others: Clustering, Information theory, Spectral analysis (Wavelets)
- Training mode
  - Supervised: different labels for normal and anomalous events
  - Semi-supervised: unlabelled dataset consisting only of normal events
  - Unsupervised: unlabelled dataset that may contain anomalies