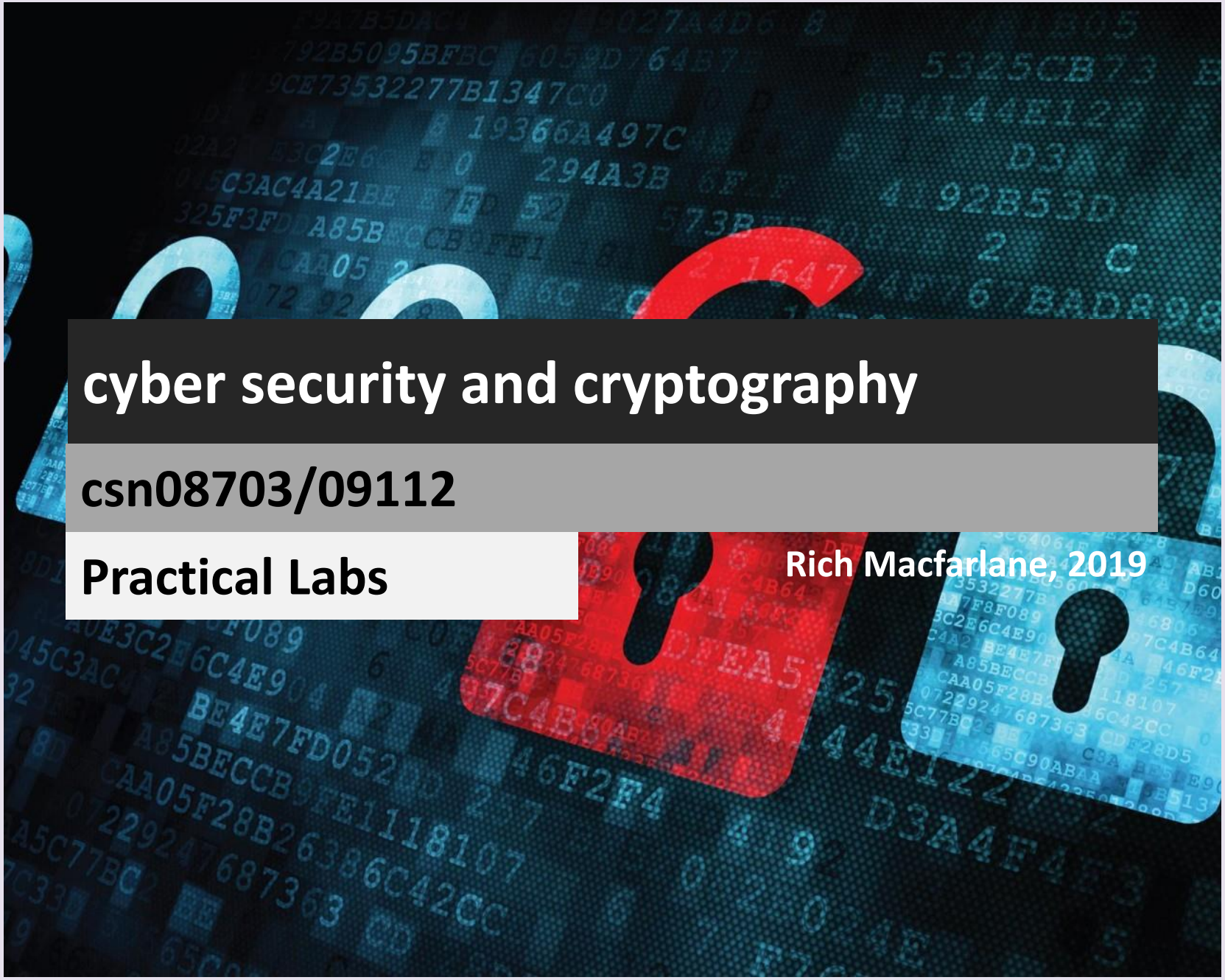


cyber security and cryptography

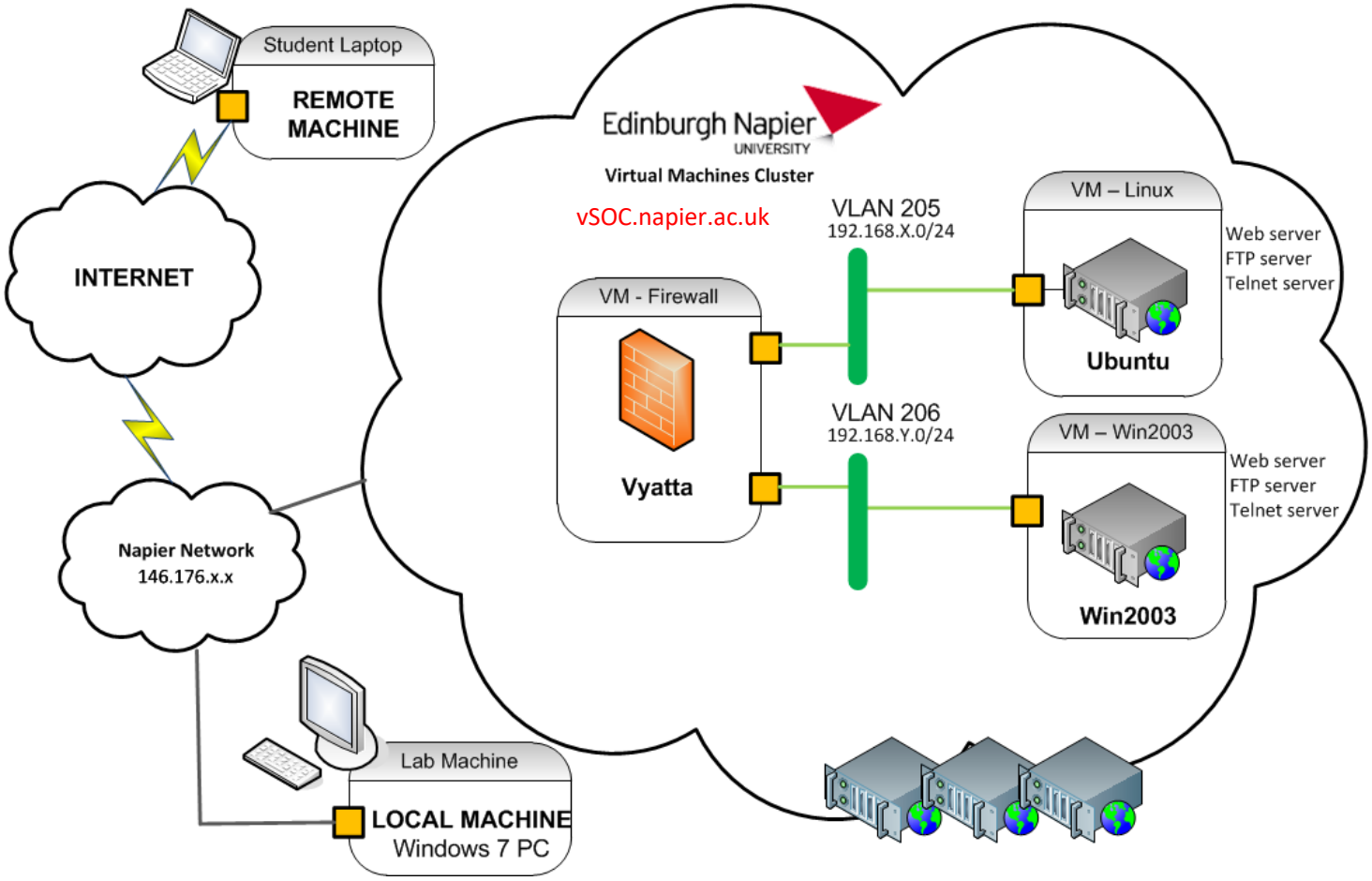
csn08703/09112

Practical Labs

Rich Macfarlane, 2019

Practical Labs – Perimeter Network

- Vyatta Firewall – Wireshark on DMZ Server



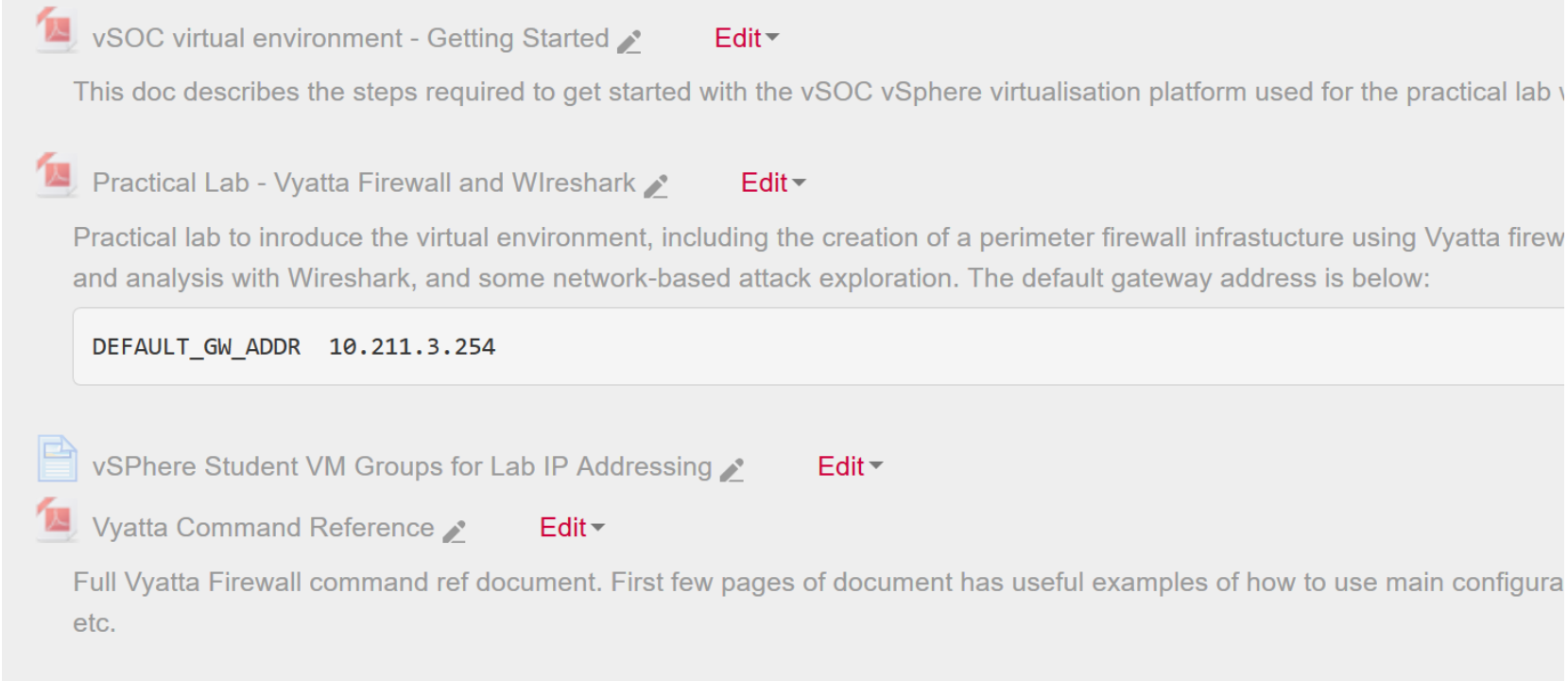
Practical Labs

Practical Labs

- Aim to develop practical skills, building understanding by exploration of authentic virtual environment
- Examples of theory, using real word tools
- Skills mainly towards coursework assessment, as well as future modules
- Keep a Log Book towards using for practical assessment
  - Paper Notebook/Electronic Document with Screen Shots
  - Print out or download pdf Labs and annotate

Practical Labs

Remotely accessible vSOC vSphere virtualised environment – read getting started pages



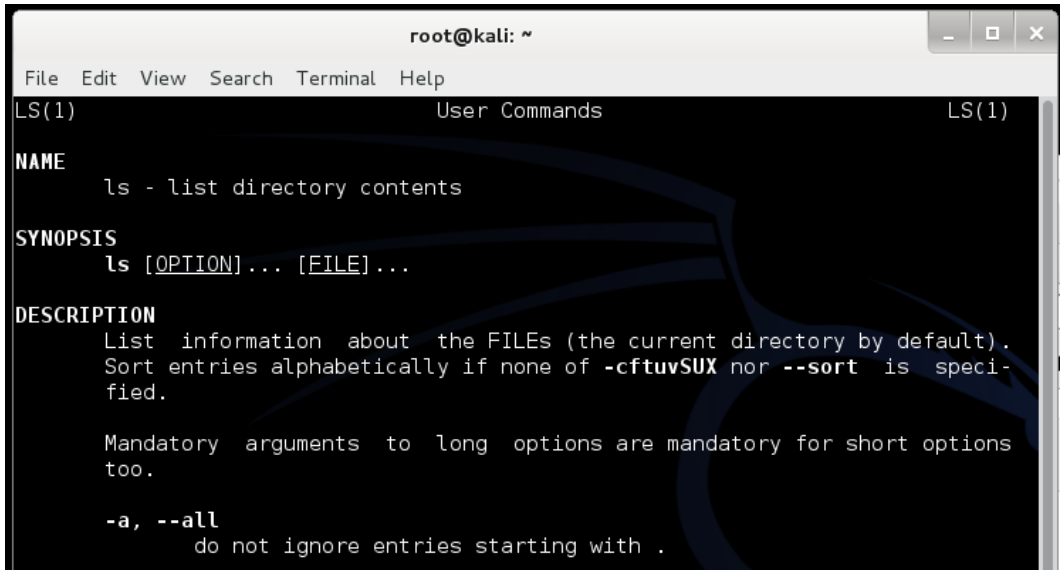
Practical Labs

- **Command Line Commands/Tools**

Using Tools (especially security tools)

- Look at tool and understand what it does FIRST… **man page**
- Only then run tool
- Do not run tool if you do not know what it does!

Check the options of the ls command by viewing the man page for the command:



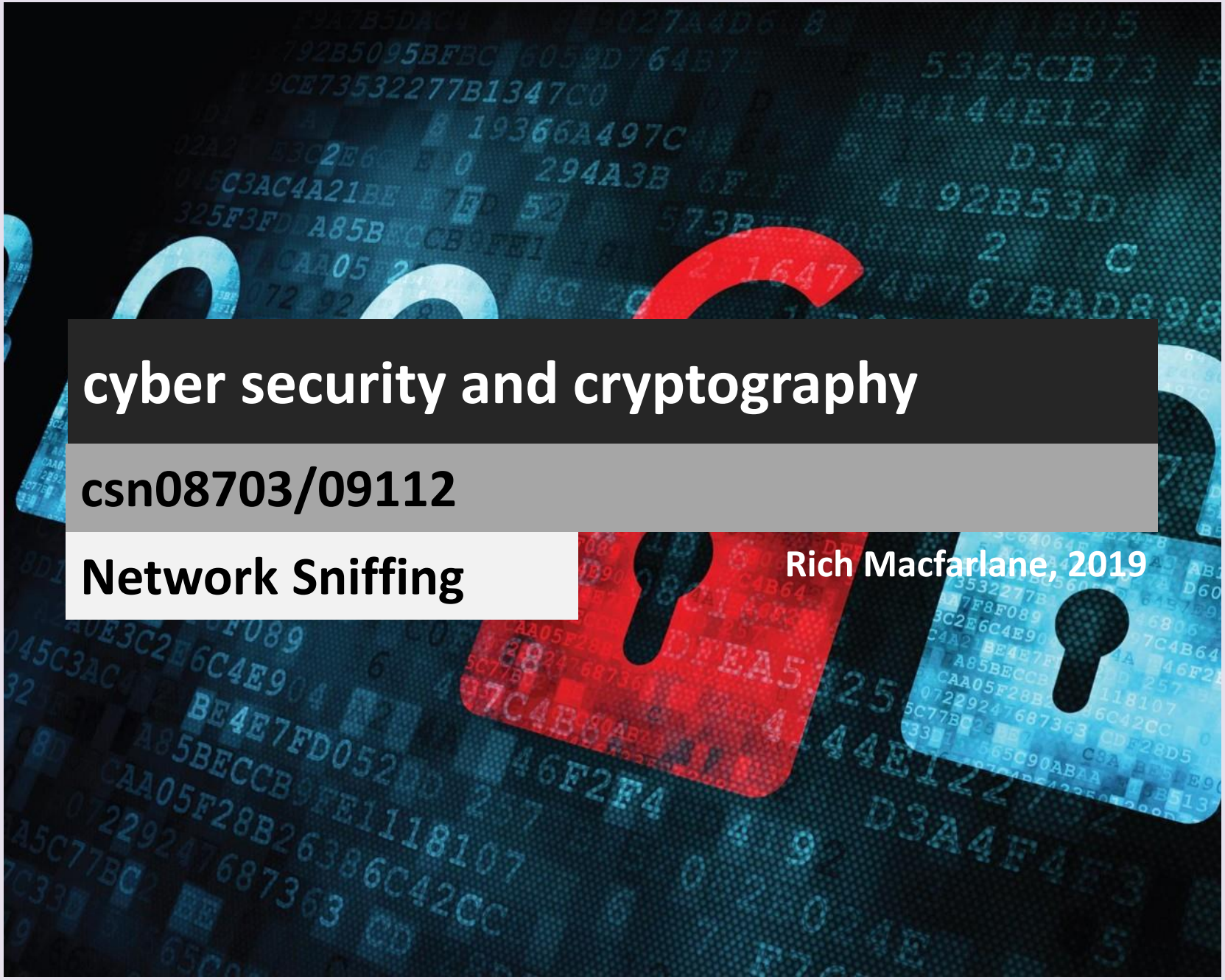
Note: f – forward a page, b – back a page, q - quit

Labs – OS Basic Commands

- Windows/Linux basic commands
  - Get help with commands
    - Linux: man page : cmd line, -h –help flags
    - Windows: `/?` Flag
  - Filesystem
    - Lixux: `pwd, cd, ls`
    - Windows: `cd, dir`
  - File content
    - Lixux: `cat, less`
    - Windows: `type`
  - Filtering content
    - Lixux: `grep, redirecting output > <`
    - Windows: `findstr`

Lab s – OS – Networking Config

- Windows/Linux networking
  - Network Configuration
    - Linux: `ip addr, ifconfig, ip route, route`
    - Windows: GUI, `ipconfig, route –a`
  - Network Connectivity
    - Linux: `ping, traceroute`
    - Windows: `ping, tracert`
  - Viewing Network Services
    - Linux: `netstat, ss`
    - Windows: `netstat -a`
  - Network Traffic Creation
    - `hping, netcat` – craft packets, mimic clients/servers



cyber security and cryptography

csn08703/09112

Network Sniffing

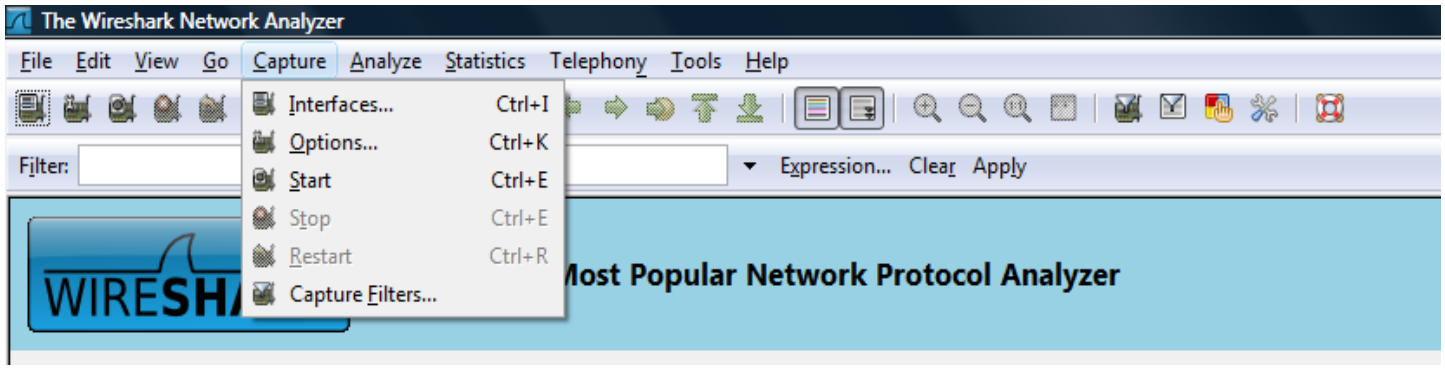
Rich Macfarlane, 2019

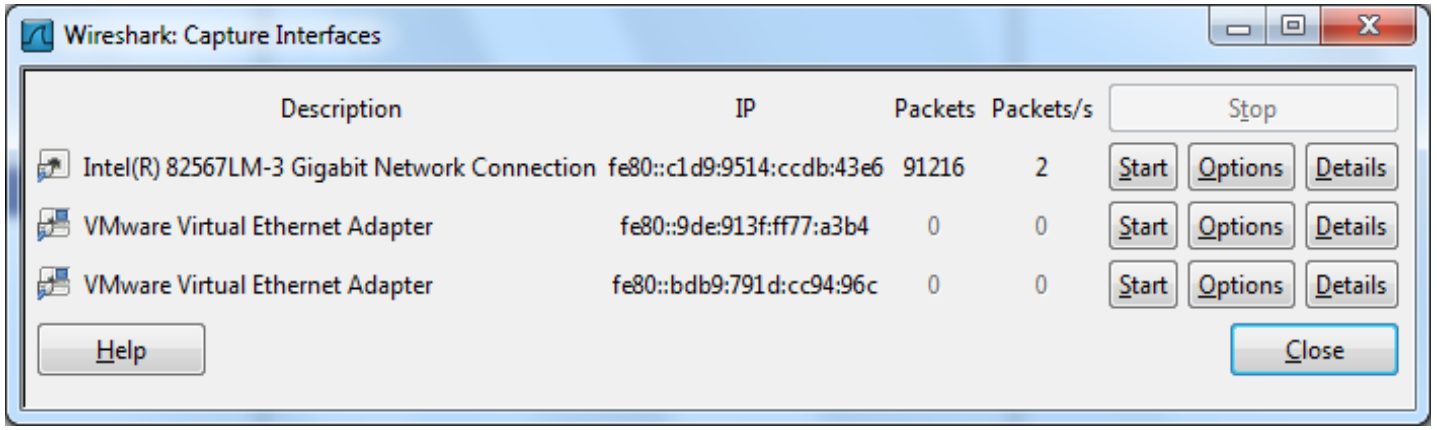
Wireshark Packet Sniffing/Analysis

- Wireshark
  - Free industry standard packet capture/analysis tool
  - Powerful built in analysis tools
    - Filtering – capture and display
    - Statistics
    - Protocol specific analysis – rebuilding streams
    - Pcap files - format can be read/witten by many tools
  - Download malware/attack pcap's and analyse

Wireshark Packet Sniffing/Analsysis

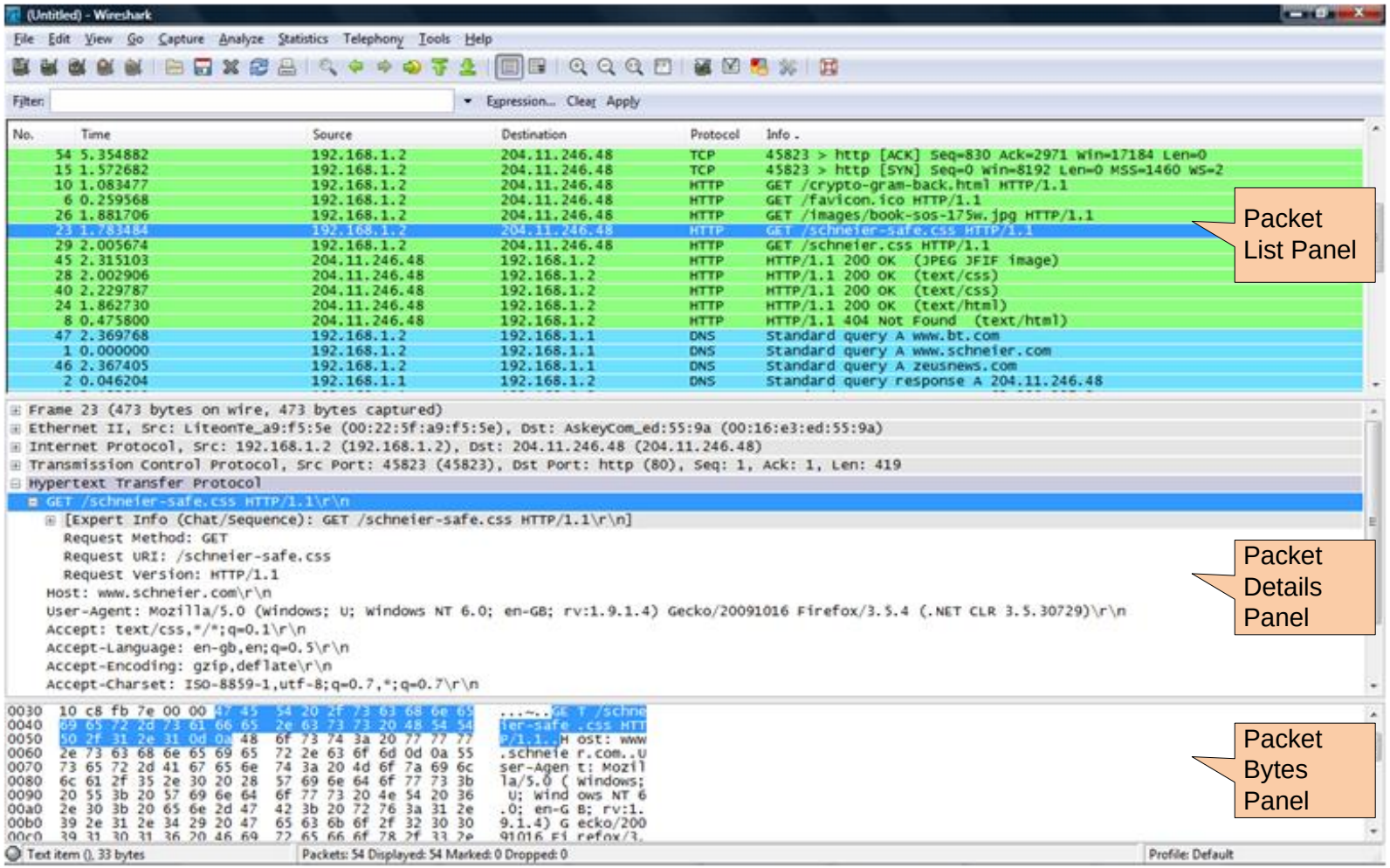
- Wireshark – Capturing on an Interface





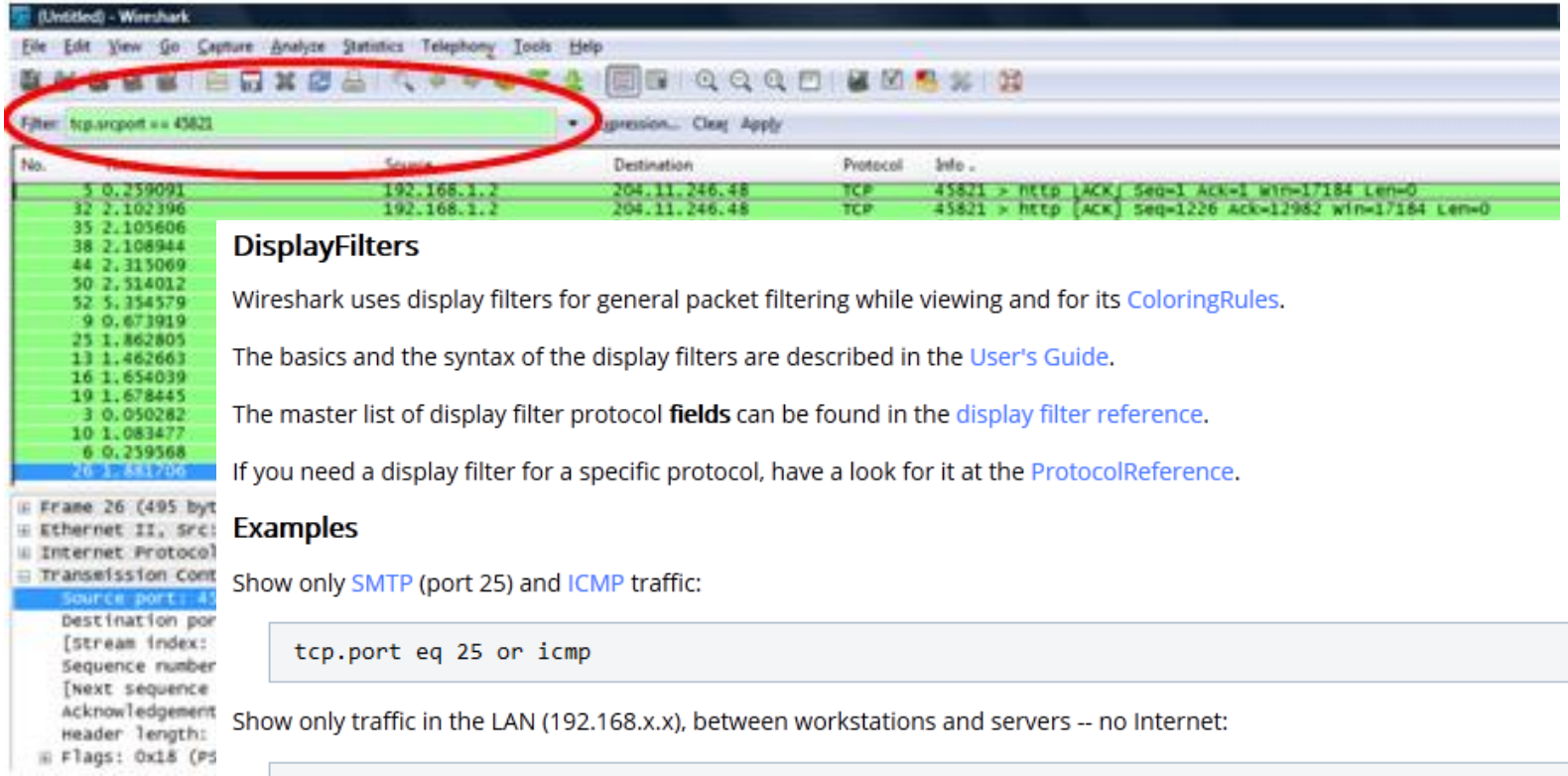
Wireshark Packet Sniffing/Analsysis

- Wireshark - Panels



Wireshark Packet Sniffing/Analsysis

- Wireshark – Display Filters



DisplayFilters

Wireshark uses display filters for general packet filtering while viewing and for its ColoringRules.

The basics and the syntax of the display filters are described in the User's Guide.

The master list of display filter protocol **fields** can be found in the display filter reference.

If you need a display filter for a specific protocol, have a look for it at the ProtocolReference.

Examples

Show only SMTP (port 25) and ICMP traffic:

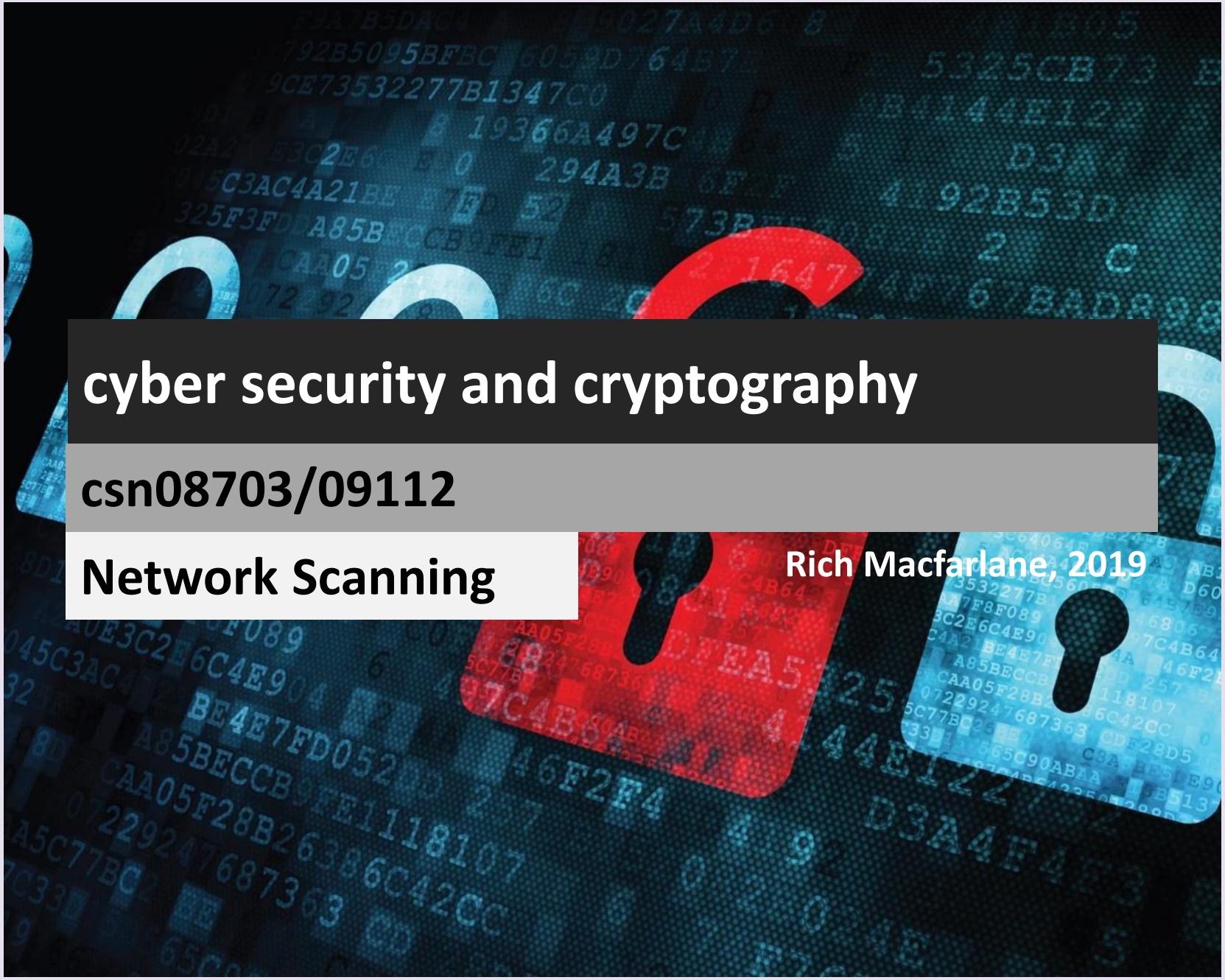
```
tcp.port eq 25 or icmp
```

Show only traffic in the LAN (192.168.x.x), between workstations and servers -- no Internet:

```
ip.src==192.168.0.0/16 and ip.dst==192.168.0.0/16
```

TCP buffer full -- *Source is instructing Destination to stop sending data*

```
tcp.window_size == 0 && tcp.flags.reset != 1
```



cyber security and cryptography

csn08703/09112

Network Scanning

Rich Macfarlane, 2019

Nmap Port Scanner

Nmap is 'the' Network Scanner

- Configurable scan types + spoofing, fragmentation
- Host sweeping
- Port scanning
- Operating System Fingerprinting
- Network Tracing
- Vulnerability Scanning with the Nmap Scripting Engine (NSE)
- Hacking Brains in Elysium!

http://nmap.org/movies

Nmap

Nmap

- Help Page: **`nmap -h`**

  Split into sections on various types of scanning (pipe into less or file)

```
Nmap 6.25 ( http://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
```

Nmap - Host Discovery

Nmap Host Discovery

`nmap -sP -n` *`target_IP_Address(s)`*

-sP: only perform host discovery, not portscan (latest **-sn**)

-n: Don't do DNS resolution - use for all scanning tools

```
root@kali:~# nmap -sn scanme.nmap.org -n

Starting Nmap 6.25 ( http://nmap.org ) at 2014-02-10 08:55 EST
Nmap scan report for scanme.nmap.org (74.207.244.221)
Host is up (0.00010s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.01 seconds
```

www.insecure.org/nmap

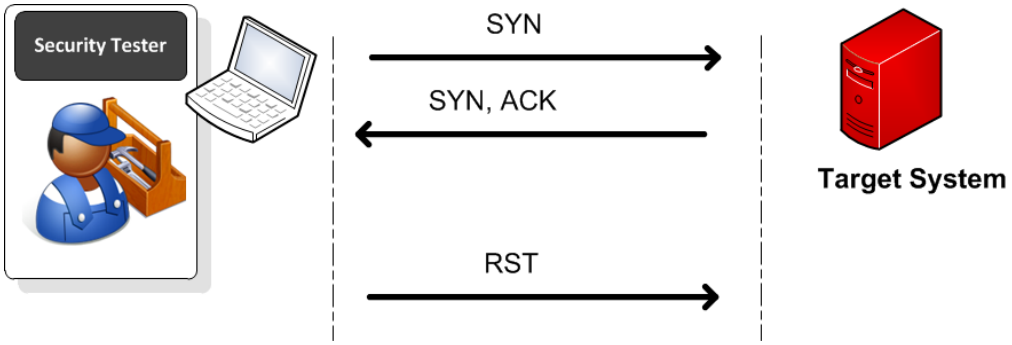
Nmap TCP Scan Types

TCP SYN Scan:

- 'Half Open' or 'SYN Stealth' Scan

  `nmap -sS target_IP_Address`

  - Does not completes the 3-way TCP handshake
  - Faster scan, and non-completed connection less likely to be logged
  - **Default scan** if nmap running with root privileges
- Responses:
  - SYN-ACK: Port Open, RST: Port Closed, ICMP or No response: Port Filtered

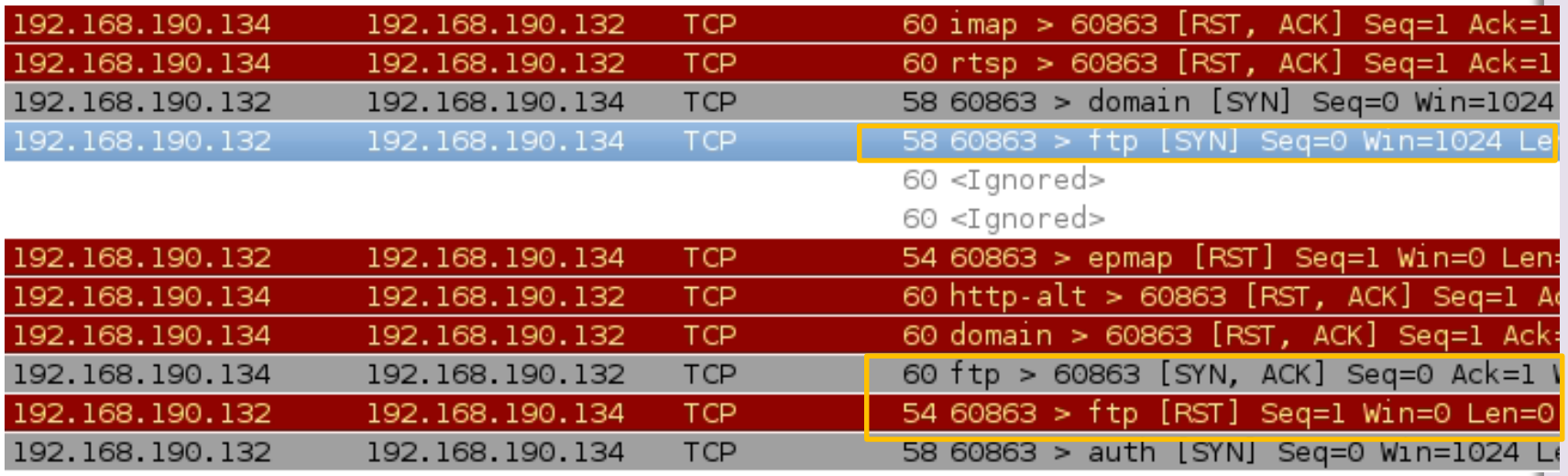


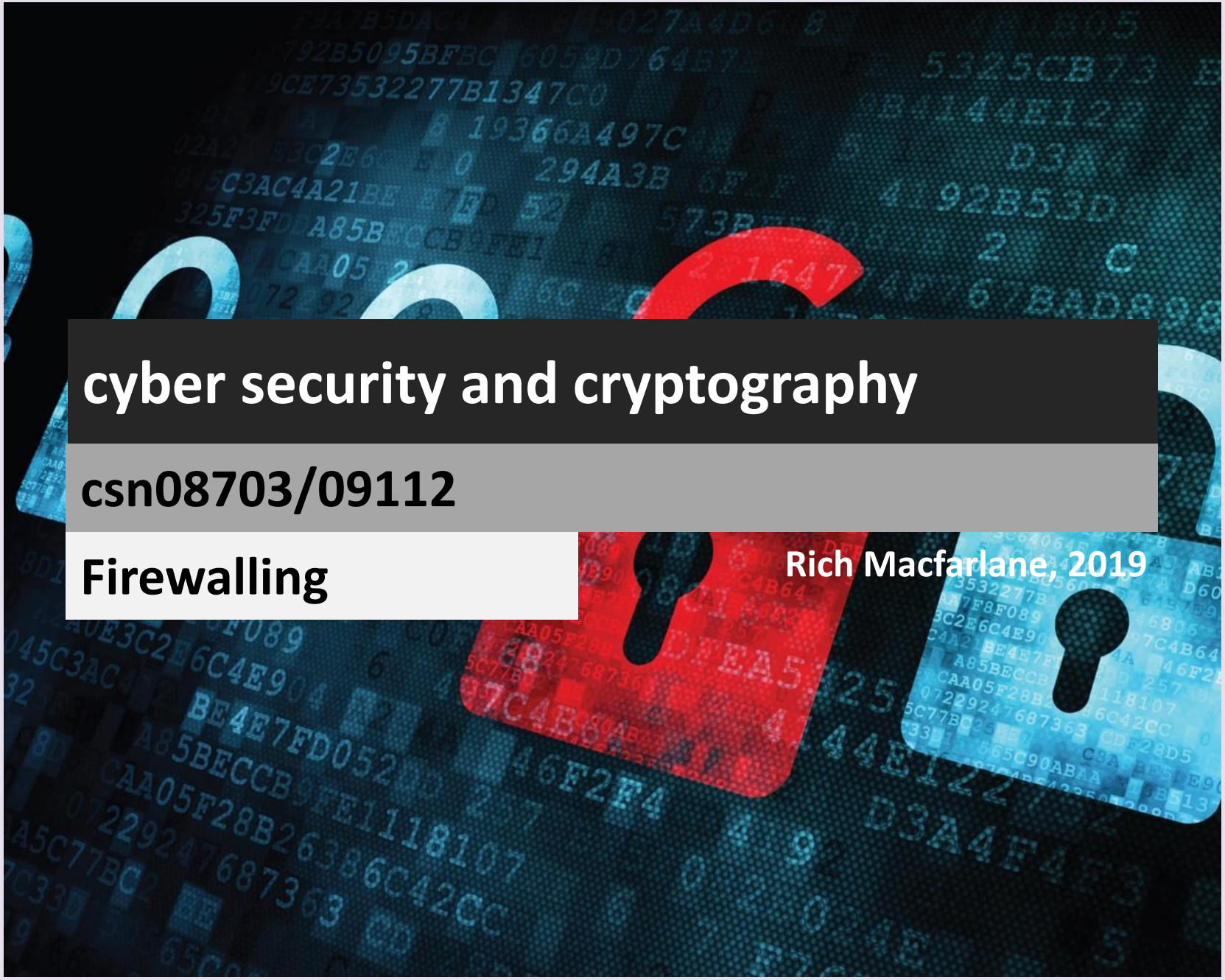
Nmap TCP Scan Types

TCP SYN Scan:

`nmap -sS` ***`target_IP_Address`***

- Open Ports reply with SYN-ACK, and then connection RST



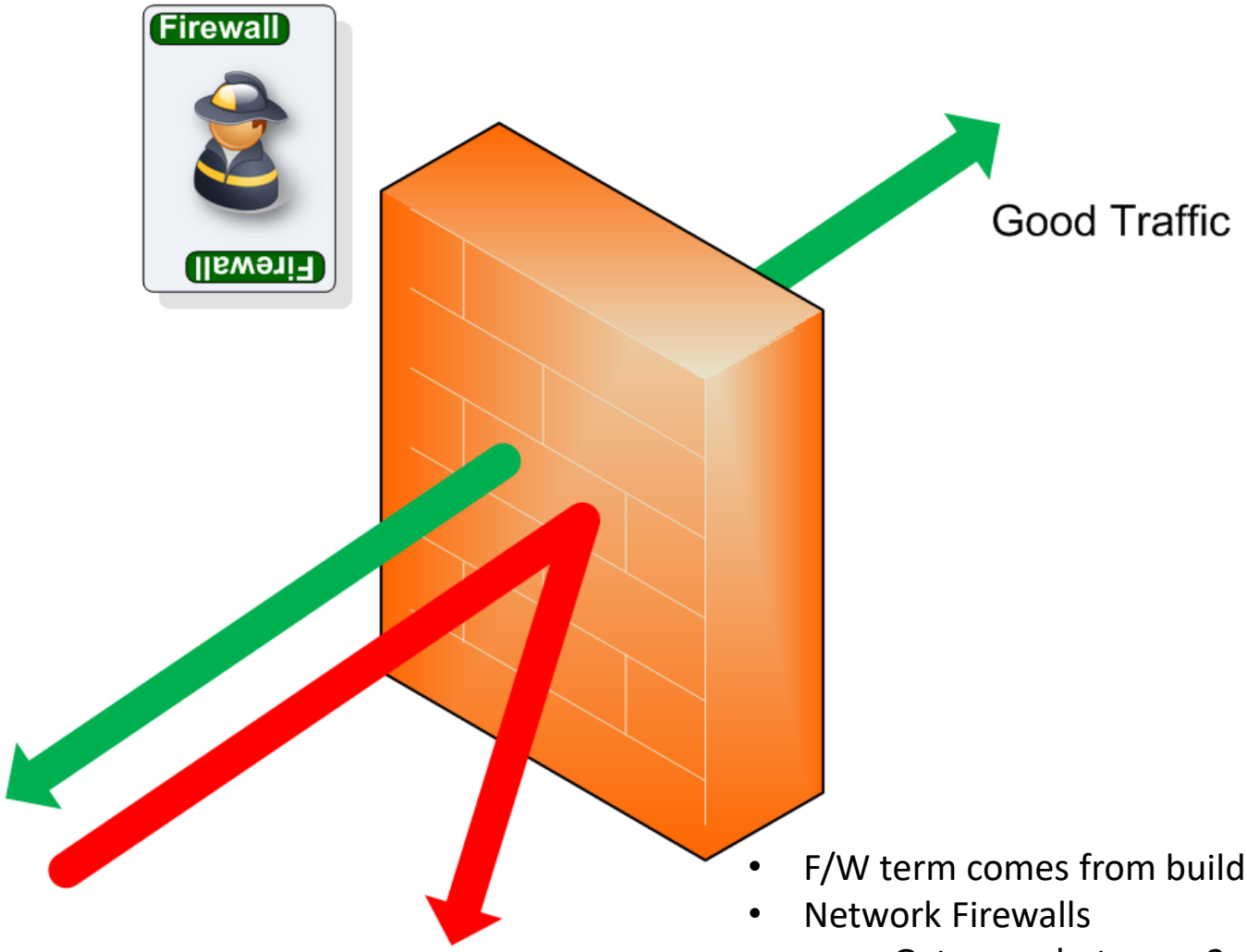


cyber security and cryptography

csn08703/09112

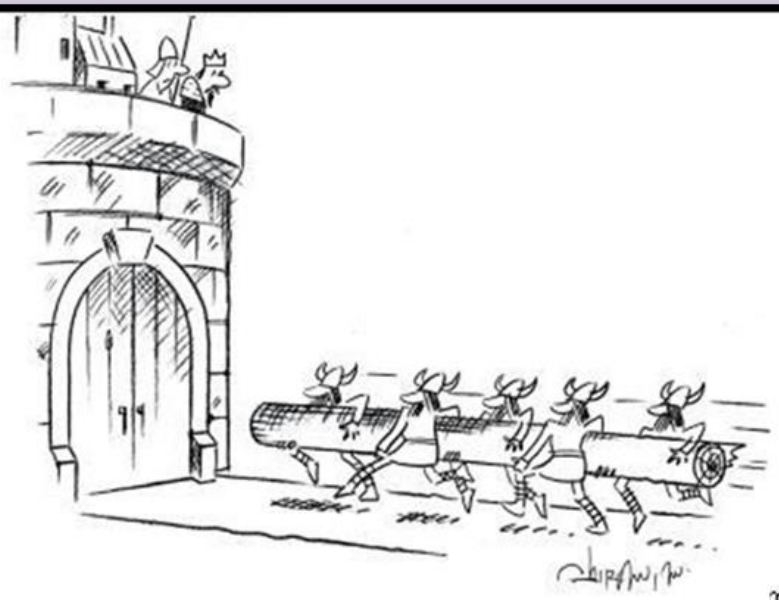
Firewalling

Rich Macfarlane, 2019

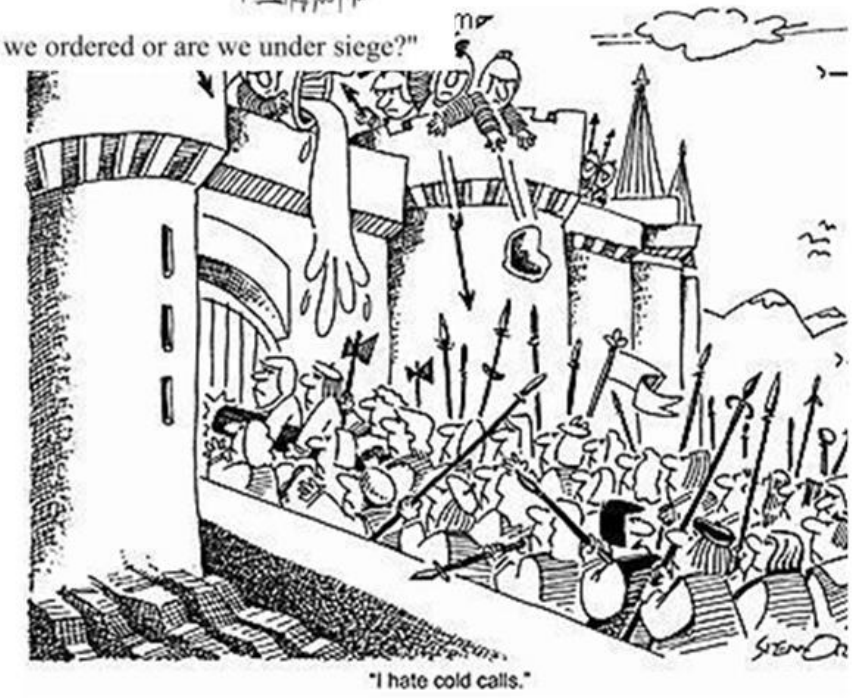


- F/W term comes from buildings
- Network Firewalls
  - Gateways between 2 networks
  - Allow Good traffic to pass
  - Reject Bad traffic

Author: Rich Macfarlane



"Is that the firewood we ordered or are we under siege?"



"I hate cold calls."

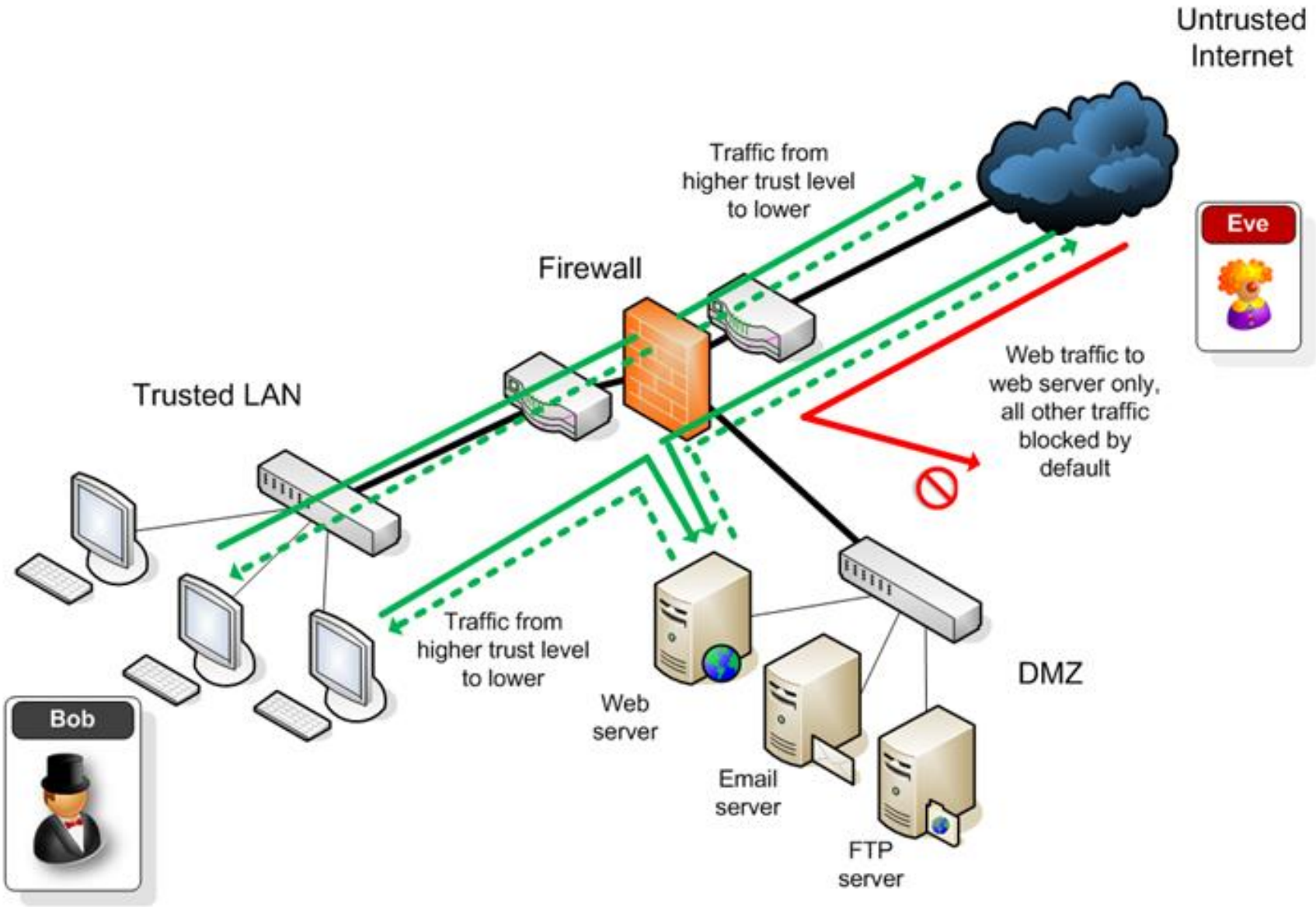
Author: Rich Macfarlane

Firewall is like a medieval castle gate

- Walls funnel people to the gate
- Guard on the gate - decides who passes through
- Choke point – all traffic should pass here

Lab 1 – Vyatta Firewall

- Vyatta - CLI
  - Command modes
    - Operational command mode
      - Help
        - `help | more`
        - `show ?`
          - `show configuration/interfaces`
    - Configuration command mode
      - `configure`
        - `set … system/interfaces/services …`
        - `commit`
        - `save`
      - `exit`



Firewall DMZ Traffic Flow

Lab 1 – Vyatta Firewall Config

- Vyatta - CLI
  - Zone-based Firewalling

Zones

```
set  zone-policy  zone  private  description  "Inside”
set  zone-policy  zone  private interface  eth1
set  zone-policy  zone  dmz description  "DMZ”
set  zone-policy  zone  dmz interface  eth2
```

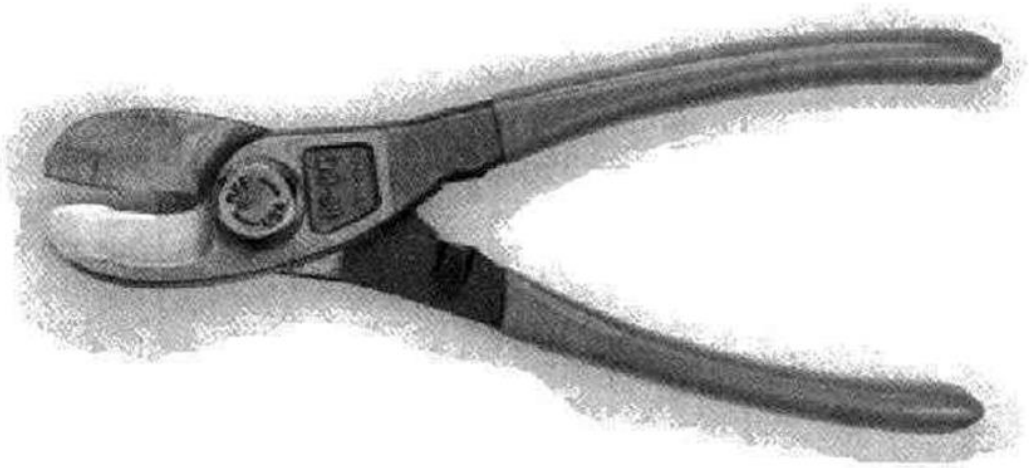
Firewall Rulesets

```
set  firewall  name  private2dmz description  “Private to DMZ"
set  firewall  name  private2dmz rule  10  action  accept
set  firewall  name  private2dmz rule  10  destination  port  80,443
set  firewall  name  private2dmz rule  10  protocol tcp

set  firewall  name  dmz2private description  "DMZ to private"
set  firewall  name  dmz2private rule  1  action  accept
set  firewall  name  dmz2private rule  1  state  established  enable
set  firewall  name  dmz2private rule  1  state  related enable
```

Rulesets applied to Zone Pairs

```
set  zone-policy  zone  private from  dmz firewall  name  dmz2private
set  zone-policy  zone  dmz from  private firewall  name  private2dmz

(review and fix any issues)
commit
```

https://wiki.vyos.net/wiki/User_Guide



The ULTIMATELY Secure Firewall

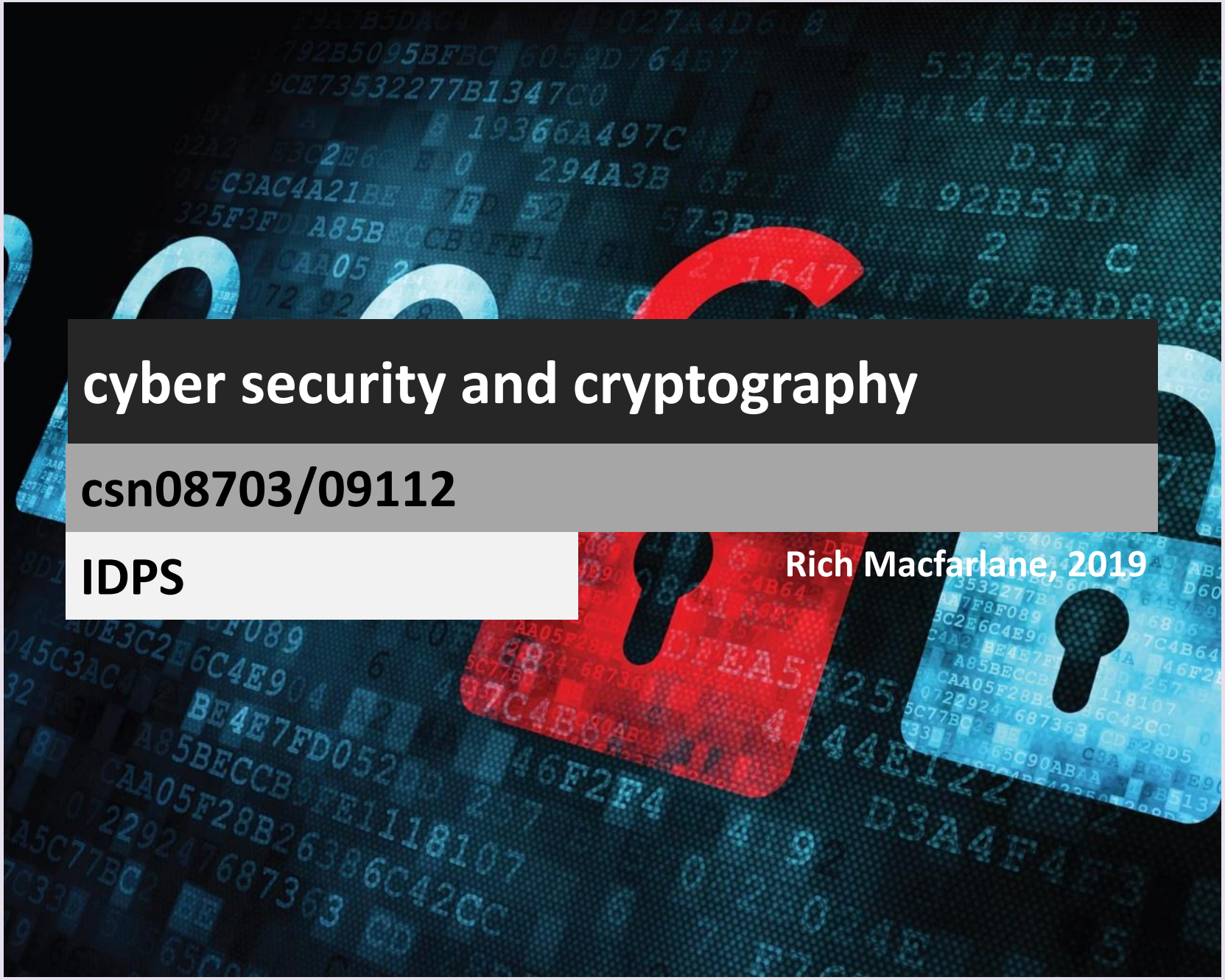
For Internet use install the firewall between the demarc of the T1 to the Internet. Place the jaws of the firewall across the T1 line lead, and bear down firmly. When your Internet service provider's network operations center calls to inform you that they have lost connectivity to your site, the firewall is correctly installed.

The fact is, that if you're connecting your network to anything else, you're running a risk. Period. Usually, that risk can be reduced, often dramatically, by employing basic security precautions such as firewalls. But a firewall is a risk reduction system, it is not a risk mitigation system -- there is, always, some danger that something can go fatally wrong with anything built by humans.

The firewall above is the only 100% guaranteed secure solution.

Marcus J. Ranum

Author: Rich macfarlane



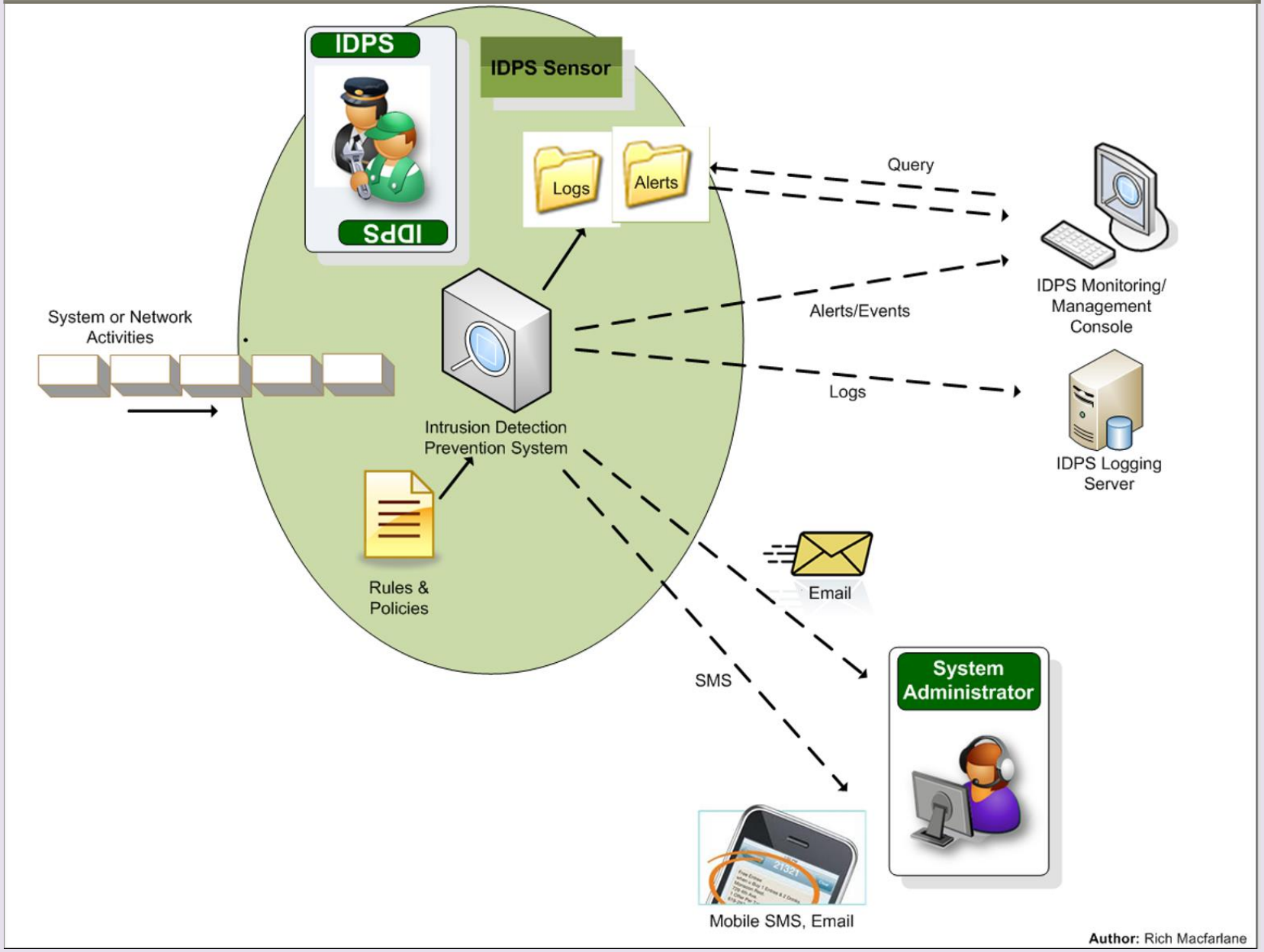
cyber security and cryptography

csn08703/09112

IDPS

Rich Macfarlane, 2019

Intrusion Detection Systems (IDS)

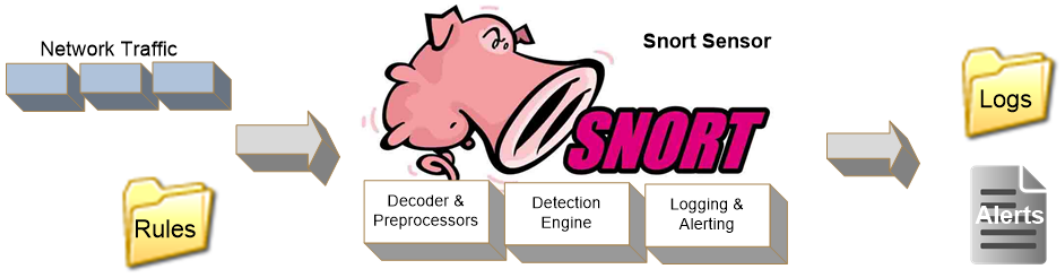


Snort IDS Rules

Snort IDS Rules have 2 Parts

- Rules Header
  - Action – What to do if rule matches packet(s) **log, alert**
  - Protocol to match - **IP, TCP, UDP, ICMP**
  - Src and Dest IP Addresses/Ranges, Ports, Direction of Packets

- Rules Options
  - Keywords/Patterns to match on in contents
  - How/Where to match in the packet
  - Alert to output when rule matches packet



Rich Macfarlane

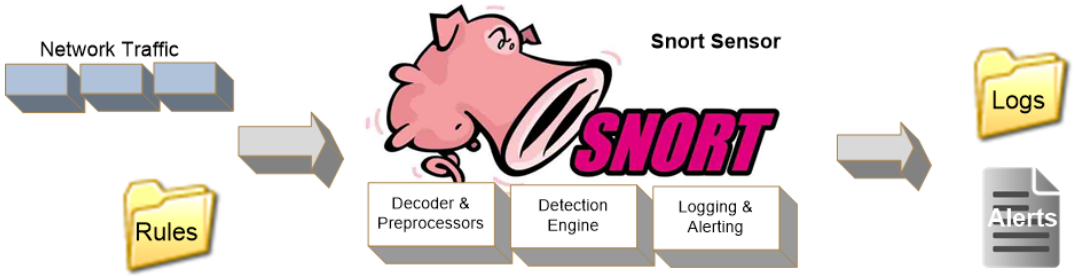
Snort IDS Rules

Snort IDS Rules have 2 Parts

- Rules Header
  - Action – **log, alert, pass**
  - Protocol to match - **IP, TCP, UDP, ICMP**
  - Src and Dest IP Addresses/Ranges, Ports, Direction of Packets
    - IP Address: **any, 1.2.3.4/netmask, !1.2.3.4/netmask, [1.2.3.4, 5.6.7.8]**
    - Ports: **any,** single port: **80,** well known ports:**1-1024, !80**
    - Direction: **->,** or **<-**

```
<Action> <Proto> <srcIP> <srcport> -> <dstIP> <dstport>
alert ip any any -> 10.1.200.7 any
alert tcp 10.1.200.7/24 -> any 22
```

```
var $WEB_SERVER 10.1.200.7
alert tcp any any -> $WEB_SERVER 80
```



Rich Macfarlane

Snort IDS Rules

Snort IDS Rules have 2 Parts

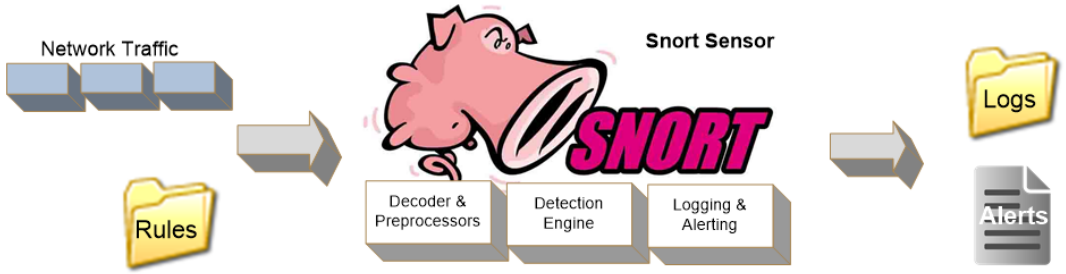
- Rules Options
  - Keywords/Patterns to match on in contents - PAYLOAD
    - **content**:"*string* |*hex*| to match"; **nocase**;
  - How/Where to match

    offset: *bytes into payload to start matching;* **depth**: *bytes to stop matching;*

    uricontent: *URL content match;* **isdataat**: *n, rel* check for data at payload
  - Alert Message to output when match happens

    `sid: rule_no, rev: revision, msg: "something detected", reference: link`

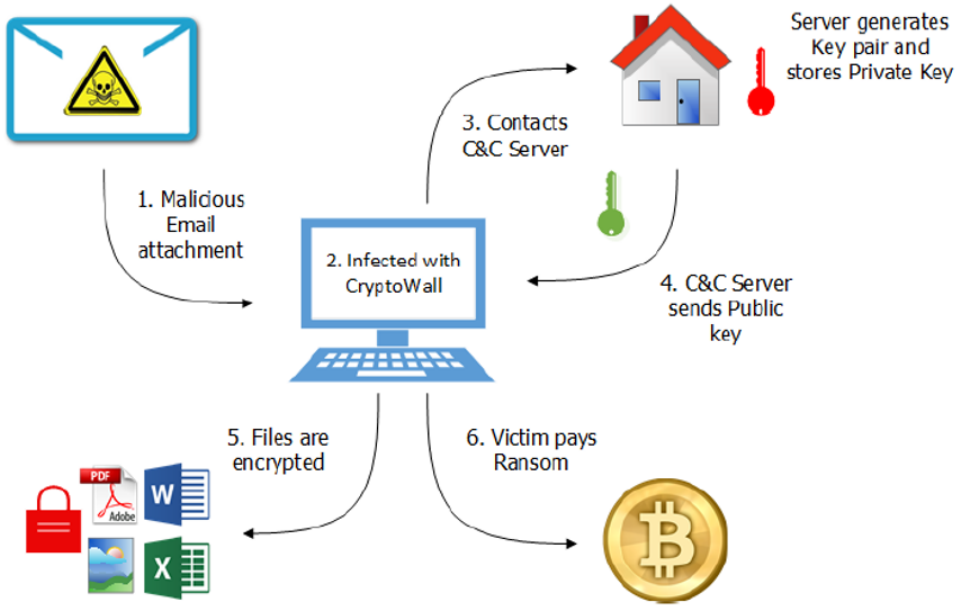
```
( <whatToMatchInPayload>; <howWhereInPayload>; <matchAlertMsg>; sid:<ruleno> )
(content:"%&'()*+"; msg:"ICMP Packet from a Linux System"; sid:1002;)
(content:"|ABCDABCD|"; msg:"ICMP packet from a Cisco Device"; sid:1001;)
```



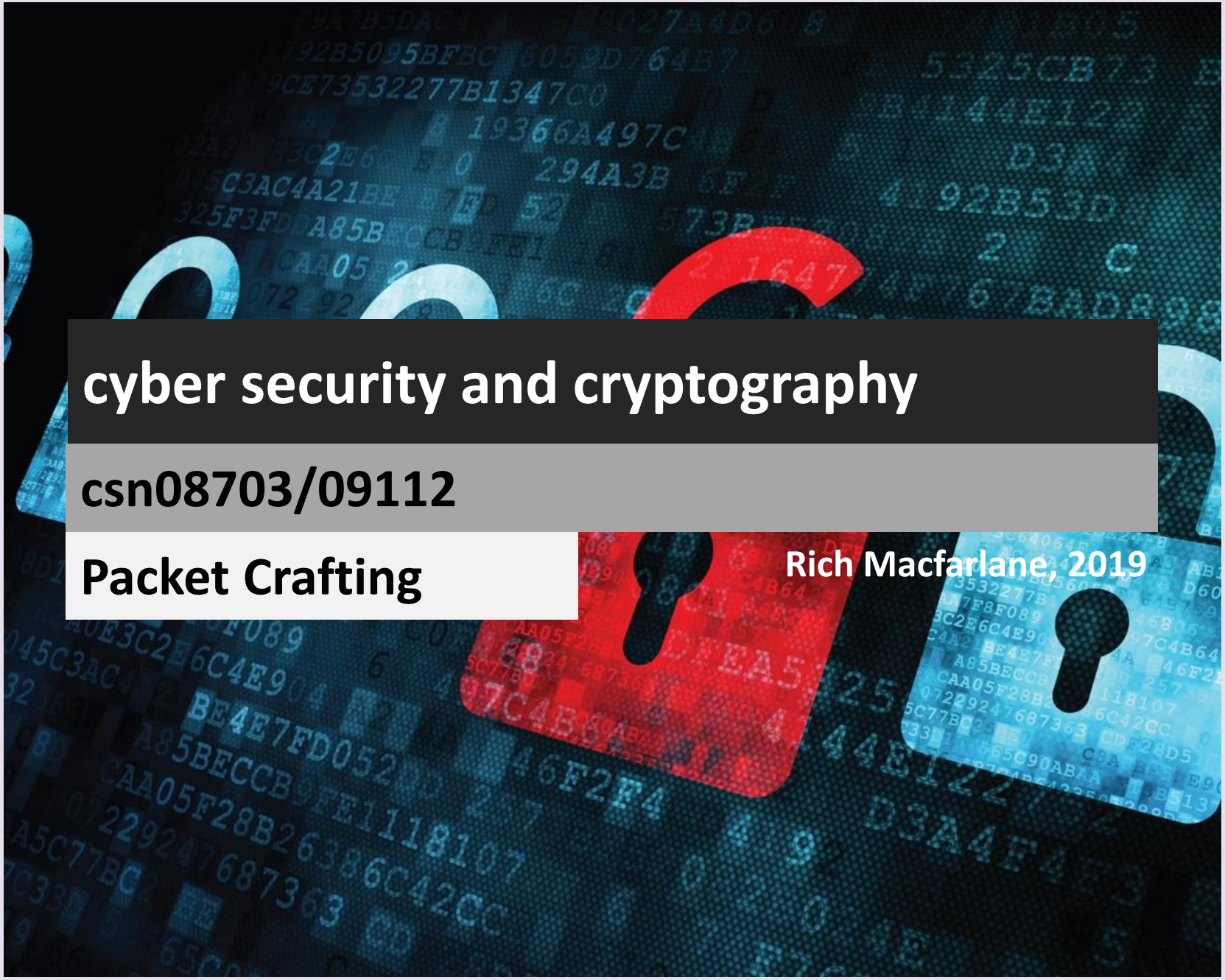
Rich Macfarlane

Snort Rule Examples

Locky Ransomware detection




```
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"MALWARE-CNC
Win.Trojan.Locky variant outbound connection"; flow:to_server,established; urilen:14;
content:"/data/info.php"; fast_pattern:only; http_uri; content:"x-requested-with:
XMLHttpRequest"; http_header; content:"Referer|3A| http|3A|"; http_header;
content:"/data"; within:25; http_header; metadata:impact_flag red, policy balanced-ips
drop, policy security-ips drop, ruleset community, service http;
reference:url,www.virustotal.com/en/file/f29ce76169727ff5a43ef7baa5c4e04f7d3302189e
3d2a31cfc9dec39e84ad03/analysis/; classtype:trojan-activity; sid:40011; rev:2;
```



cyber security and cryptography

csn08703/09112

Packet Crafting

Rich Macfarlane, 2019

Hping – Packet Crafting

Hping

- Can be used to craft and send any TCP/IP packets to host(s)
- Host discovery, Port scanning, Mimicking attacks
- Default Protocol is TCP packets, with no TCP flags set, to port 0
  - Typically machines reply with a RST, identifying a target

- **`hping3 10.200.0.1 -c 2`**

```
root@host-7-129:~# hping3 10.200.0.1 -c 2
HPING 10.200.0.1 (eth0 10.200.0.1): NO FLAGS are set, 40 headers + 0 data bytes
```

```
16:42:22.084681 IP (tos 0x0, ttl 64, id 3520, offset 0, flags [none], proto TCP (6), length 40)
    10.0.7.129.1582 > 10.200.0.1.0: Flags [none], cksum 0x151a (correct), win 512, length 0
```

- Craft packets to use other Protocols:

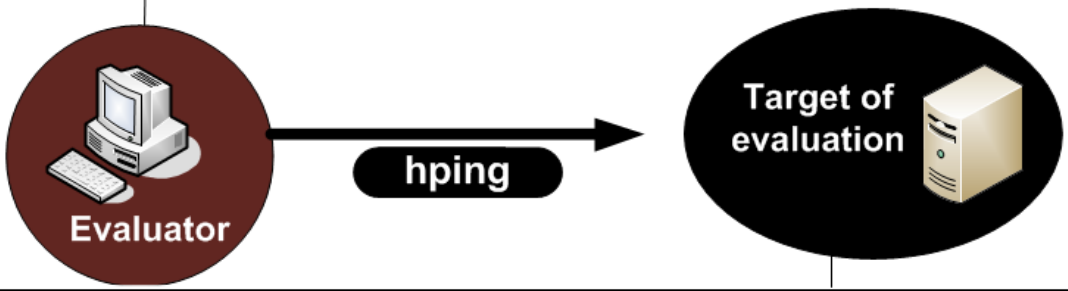
--udp	send UDP packets
--icmp	send ICMP packets
--rawip	send IP packets (no TCP or UDP header)

`hping3 --icmp www.google.com -c 2`

```
root@kali:~# hping3 --icmp www.google.com
HPING www.google.com (eth0 173.194.34.178): icmp mode set, 28 headers
len=46 ip=173.194.34.178 ttl=128 id=32884 icmp_seq=0 rtt=60.1 ms
```

```
04:51:21.471981 IP 173.194.34.178 > 192.168.23.135: ICMP echo reply, id 13102, seq 256,
        0x0000:  4500 001c 8075 0000 8001 11c8 adc2 22b2  E....u........".
        0x0010:  c0a8 1787 0000 cbd1 332e 0100 0000 0000  ........3.......
        0x0020:  0000 0000 0000 0000 0000 0000 0000       ..............
```

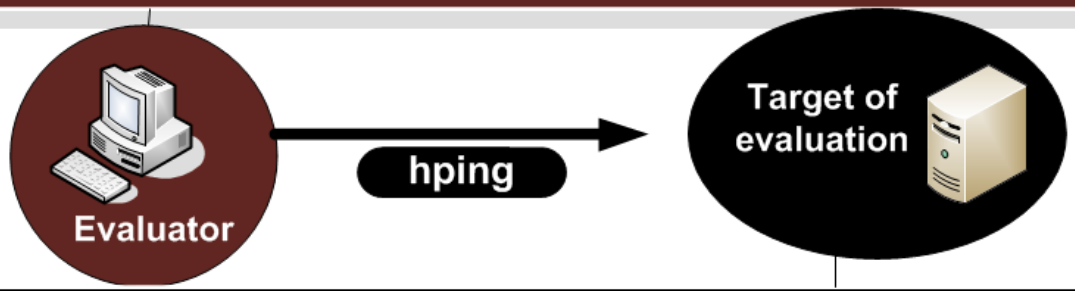
```
napier@ubuntu:~$ sudo hping -S 192.168.75.132 -e eth0
[sudo] password for napier:
HPING 192.168.75.132 (eth0 192.168.75.132): S set, 40 headers + 4 data bytes
[main] memlockall(): Success
Warning: can't disable memory paging!
len=46 ip=192.168.75.132 ttl=128 id=2052 sport=0 flags=RA seq=0 win=0 rtt=69.3 ms
len=46 ip=192.168.75.132 ttl=128 id=2053 sport=0 flags=RA seq=1 win=0 rtt=0.5 ms
len=46 ip=192.168.75.132 ttl=128 id=2054 sport=0 flags=RA seq=2 win=0 rtt=8.9 ms
--- 192.168.75.132 hping statistic ---
7 packets transmitted, 7 packets received, 0% packet loss
```




```
14:03:05.859738 IP ubuntu.local.2714 > 192.168.75.132.0: Flags [S], seq
  1222983093:1222983097, win 512, length 4
14:03:05.859975 IP 192.168.75.132.0 > ubuntu.local.2714: Flags [R.], seq 0, ack
  1222983098, win 0, length 0
14:03:06.860566 IP ubuntu.local.2715 > 192.168.75.132.0: Flags [S], seq
  1026211710:1026211714, win 512, length 4
```

Author: Prof Bill Buchanan

```
napier@ubuntu:~$ sudo hping -S 192.168.75.132 -e eth0 -p 80
HPING 192.168.75.132 (eth0 192.168.75.132): S set, 40 headers + 4 data bytes
[main] memlockall(): Success
Warning: can't disable memory paging!
len=46 ip=192.168.75.132 ttl=128 id=2072 sport=80 flags=SA seq=0 win=64240 rtt=11.3
ms
len=46 ip=192.168.75.132 ttl=128 id=2073 sport=80 flags=SA seq=1 win=64240 rtt=0.5
ms
len=46 ip=192.168.75.132 ttl=128 id=2074 sport=80 flags=SA seq=2 win=64240 rtt=0.4
ms
--- 192.168.75.132 hping statistic ---
15 packets transmitted, 15 packets received, 0% packet loss
round-trip min/avg/max = 0.4/1.5/11.3 ms
```




```
14:04:31.090418 IP ubuntu.local.2222 > 192.168.75.132.www: Flags [S], seq
  56776272:56776276, win 512, length 4
14:04:31.092037 IP ubuntu.local.57490 > 192.168.75.2.domain: 34223+ PTR?
  132.75.168.192.in-addr.arpa. (45)
14:04:31.093064 IP 192.168.75.132.www > ubuntu.local.2222: Flags [S.], seq
  447090437, ack 56776273, win 64240, options [mss 1460], length 0
14:04:31.093132 IP ubuntu.local.2222 > 192.168.75.132.www: Flags [R], seq
  56776273, win 0, length 0
```

Author: Prof Bill Buchanan