

Welcome to

Core Infrastructure and BGP Intro

Building a production network

Henrik Lund Kramshøj hlk@zencurity.com @kramse 💆 🕽

Slides are available as PDF, kramse@Github core-infrastructure.tex in the repo security-courses

Goal





Spend some hours setting up a production network:

Use VLANs, IP subnetting, routing, BGP
Use Automated provisioning scripts Ansible
Use SNMP and introduce some debugging, monitoring tools
Discuss how to Monitor, Mirror data and start an IDS
We try to do a lot, feel free to focus on specific parts

Plan for today



- Design a robust network
- Isolation and segmentation
- (Routing Security) removed, see Running a Modern Network slide set
- Switch and access security, port security
- (Wireless security) removed

Think if you could redesign your office network!

Reading Ideas



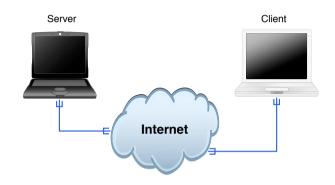
- Read
- https://nsrc.org/workshops/2018/myren-nsrc-cndo/networking/cndo/en/presentations/Campus_Security_Overview.pdf
- https://nsrc.org/workshops/2018/tenet-nsrc-cndo/networking/cndo/en/presentations/Campus_Operations_BCP.pdf
- Download, but dont read it all

https://nsrc.org/workshops/2015/apricot2015/raw-attachment/wiki/Track1Agenda/01-ISP-Network-Design.pdf

These are good resources when you come home, and want to build networks!

Internet today





Clients and servers
Roots in academia
Protocols more than 20 years old
HTTP is becoming encrypted, but a lot other traffic is not

OSI og Internet modellerne



OSI Reference Model

Application

Presentation

Session

Transport

Network

Link

Physical

Internet protocol suite

Applications	NFS
HTTP, SMTP, FTP,SNMP,	XDR
	RPC
TCP UDP	
IPv4 IPv6 I	CMPv6 _{ICMP}
ARP RARP MAC	
Ethernet token-ring ATM	

Design a robust network Isolation and segmentation

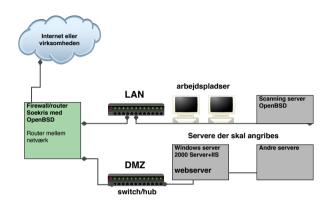


Hvad kan man gøre for at få bedre netværkssikkerhed?

- Bruge switche der skal ARP spoofes og bedre performance
- Opdele med firewall til flere DMZ zoner for at holde udsatte servere adskilt fra hinanden, det interne netværk og Internet
- Overvåge, læse logs og reagere på hændelser
 - Husk du skal også kunne opdatere dine servere

Basic Network Security Pattern Isolate in VLANs





Du bør opdele dit netværk i segmenter efter trafik Du bør altid holde interne og eksterne systemer adskilt! Du bør isolere farlige services i jails og chroots Brug port security til at sikre basale services DHCP, Spanning Tree osv.

Our Networks



We will now configure networks, using our sample switch TP-Link T1500G-10PS

Core network provides uplink through a switch / internet exchange Each team will need:

- A switch TP-Link T1500G-10PS L2 features default config
- USB Ethernet or VLAN compatible virtualization network
- Ethernet cables

Exercise in networking VLANs, Routing and RPF



Each team will configure:

- Debian VM router-on-a-stick L3 forwarding https://en.wikipedia.org/wiki/One-armed_router
- Recommended to serve DHCP service, and possibly NTP etc.
- Configure Monitoring and LibreNMS optional
- Reconfigure uplink from static routing to BGP optional
- Connect your IDS optional, Configure port security optional

Use the guides from:

https://www.tp-link.com/uk/support/download/t1500g-10ps/#Related-Documents

Packet sniffing tools



Tcpdump for capturing packets

Wireshark for dissecting packets manually with GUI

Zeek Network Security Monitor

Discuss Suricata, modern robust capable of IDS and IPS (prevention), ntopng High-speed web-based traffic analysis and Maltrail Malicious traffic detection system https://github.com/stamparm/MalTrail

Often a combination of tools and methods used in practice

Full packet capture big data tools also exist

Using Wireshark



```
http-example.cap

    □ □ X

  Apply a display filter ... < %/>
                 172.24.65.102 91.102.91.18 TCP
  1 0.000000
                                                              58816 → http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM...
  2 0.000170
                 172.24.65.102 91.102.91.18
                                                              58817 - http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM.
  3 0.127053
                 91.102.91.18 172.24.65.102 TCP
                                                              http - 58816 [SYN, ACK] Seg=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK PERM=1 WS=8 TSval=18552...
  4 0.127167
                 91.102.91.18 172.24.65.102 TCP
                                                              http - 58817 [SYN. ACK] Sec=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK PERM=1 WS=8 TSval=25124...
  5 0.127181
                 172.24.65.102 91.102.91.18 TCP
                                                              58816 → http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=1855239975
  6 0.127226
                 172.24.65.102 91.102.91.18 TCP
                                                              58817 → http [ACK] Seg=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=2512433851
  8 0.141320
                 91.102.91.18 172.24.65.102 HTTP
                                                               HTTP/1.1 304 Not Modified
  9 0.141421 172.24.65.102 91.102.91.18 TCP
                                                              58816 → http [ACK] Seg=503 Ack=190 Win=131568 Len=0 TSval=745562551 TSecr=1855239975
 Frame 7: 568 bytes on wire (4544 bits), 568 bytes captured (4544 bits)
 ▶ Ethernet II, Src: Apple_6c:87:5e (7c:d1:c3:6c:87:5e), Dst: Cisco_32:09:30 (44:2b:03:32:09:30)
 ▶ Internet Protocol Version 4, Src: 172.24.65.102 (172.24.65.102), Dst: 91.102.91.18 (91.102.91.18)
 ▶ Transmission Control Protocol, Src Port: 58816 (58816), Dst Port: http (80), Seq: 1, Ack: 1, Len: 502
 Hypertext Transfer Protocol
   ▶ GET / HTTP/1.1\r\n
     Host: 91.102.91.18\r\n
     Connection: keep-alive\r\n
     Cache-Control: max-age=0\r\n
     Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8\r\n
     User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36\r\n
     Accept-Encoding: gzip,deflate,sdch\r\n
     Accept-Language: en-US,en;q=0.8,cs;q=0.6,da;q=0.4\r\n
     If-None-Match: "7053a63e31516a58b27a295edb31d07524a6e0a3"\r\n
     If-Modified-Since: Tue, 17 Nov 2009 11:22:22 GMT\r\n
     [Full request URI: http://91.102.91.18/]
     [HTTP request 1/1]
     [Response in frame: 8]
0000 44 2b 03 32 09 30 7c d1 c3 6c 87 5e 08 00 45 00 D+.2.0|Ñ Ãl.^..E.
0010 02 2a 9e d7 40 00 40 06 f5 ff ac 18 41 66 5b 66 .*.×@.@. öÿ¬.Af[f
0020 5b 12 e5 c0 00 50 08 ea 0e c7 03 14 0c 19 80 18 [.âÀ.P.ê .Ç.....
0030 20 2b 0f c0 00 00 01 01 08 0a 2c 70 61 aa 6e 94
                                                       +.À....,paªn.
-'GET / HTTP/1.1
0040 b7 27 47 45 54 20 2f 20 48 54 54 50 2f 31 2e 31
0050 0d 0a 48 6f 73 74 3a 20 39 31 2e 31 30 32 2e 39
0060 31 2e 31 38 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 1.18..Co nnection
0070 3a 20 6b 65 65 70 2d 61 6c 69 76 65 0d 0a 43 61
                                                       : keep-a live..Ca
0080 63 68 65 2d 43 6f 6e 74 72 6f 6c 3a 20 6d 61 78 che-Cont rol: max
0090 2d 61 67 65 3d 30 0d 0a 41 63 63 65 70 74 3a 20 -age=0.. Accept:
00a0 74 65 78 74 2f 68 74 6d 6c 2c 61 70 70 6c 69 63 text/htm l.applic
00b0 61 74 69 6f 6e 2f 78 68 74 6d 6c 2b 78 6d 6c 2c ation/xh tml+xml,
00c0 61 70 70 6c 69 63 61 74 69 6f 6e 2f 78 6d 6c 3b applicat ion/xml;
http-example.cap
                                                                                             Packets: 9 · Displayed: 9 · Marked: 0 · Load time: 0:0.0
```

https://www.wireshark.org

Kali Linux the pentest toolbox





Kali http://www.kali.org/ brings together 100s of tools 100.000s of videos on youtube alone, searching for kali and \$TOOL

Use this to generate bad traffic

Hackerlab setup





- Hardware: most modern laptops has CPU with virtualization
 May need to enale it in BIOS
- Software: use your favorite operating system, Windows, Mac, Linux
- Virtualization software: VMware, Virtual box, choose your poison
- Hackersoftware: Kali as a Virtual Machine https://www.kali.org/
- Install sniffing VM put into bridge mode

What happens today?



Think like a network architect team member

Get basic tools running

Start rouiting and creating a network

Improve situation, think about how to monitor while building

Today focus on the lower parts, but user interfaces are important too





Now lets do the exercise

Bonus: Download Kali Linux Revealed (KLR) Book 10 min

which is number 1 in the exercise PDF.





Now lets do the exercise

Bonus: Check your Kali VM, run Kali Linux 30 min

which is number 2 in the exercise PDF.





Now lets do the exercise

Bonus: Wireshark and Tcpdump 15 min

which is number 3 in the exercise PDF.





Now lets do the exercise

Bonus: Capturing TCP Session packets 10 min

which is number 4 in the exercise PDF.





Now lets do the exercise

Bonus: Using ping and traceroute 10 min

which is number 5 in the exercise PDF.





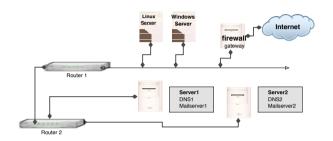
Now lets do the exercise

Bonus: DNS and Name Lookups 10 min

which is number 6 in the exercise PDF.

Network mapping





Using traceroute and similar programs it is often possible to make educated guess to network topology

Time to live (TTL) for packets are decreased when crossing a router when it reaches zero the packet is timed out, and ICMP message sent back to source

Default Unix traceroute uses UDP, Windows tracert use ICMP

traceroute - UDP



```
# tcpdump -i en0 host 10.20.20.129 or host 10.0.0.11
tcpdump: listening on en0
23:23:30.426342 10.0.0.200.33849 > router.33435: udp 12 [ttl 1]
23:23:30.426742 safri > 10.0.0.200: icmp: time exceeded in-transit
23:23:30.436069 10.0.0.200.33849 > router.33436: udp 12 [ttl 1]
23:23:30.436357 safri > 10.0.0.200: icmp: time exceeded in-transit
23:23:30.437117 10.0.0.200.33849 > router.33437: udp 12 [ttl 1]
23:23:30.437383 safri > 10.0.0.200: icmp: time exceeded in-transit
23:23:30.437574 10.0.0.200.33849 > router.33438: udp 12
23:23:30.438946 router > 10.0.0.200: icmp: router udp port 33438 unreachable
23:23:30.451319 10.0.0.200.33849 > router.33439: udp 12
23:23:30.452569 router > 10.0.0.200: icmp: router udp port 33439 unreachable
23:23:30.452813 10.0.0.200.33849 > router.33440: udp 12
23:23:30.454023 router > 10.0.0.200: icmp: router udp port 33440 unreachable
23:23:31.379102 10.0.0.200.49214 > safri.domain: 6646+ PTR?
200.0.0.10.in-addr.arpa. (41)
23:23:31.380410 safri.domain > 10.0.0.200.49214: 6646 NXDomain* 0/1/0 (93)
14 packets received by filter
O packets dropped by kernel
```

Low TTL, UDP, high ports above 33000 = Unix traceroute signature

Experiences gathered



Lots of information

Reveals a lot about the network, operating systems, services etc.

I use a template when getting data

- Respond to ICMP: □ echo, □ mask, □ time
- Respond to traceroute: □ ICMP, □ UDP
- Open ports TCP og UDP:
- Operating system:
- ... (banner information)

Beware when doing scans it is possible to make routers, firewalls and devices perform badly or even crash!

The Zeek Network Security Monitor



The Zeek Network Security Monitor

Why Choose Zeek? Zeek is a powerful network analysis framework that is much different from the typical IDS you may know.

Adaptable

Zeek's domain-specific scripting language enables sitespecific monitoring policies.

Efficient

Zeek targets high-performance networks and is used operationally at a variety of large sites.

Flexible

Zeek is not restricted to any particular detection approach and does not rely on traditional signatures.

Forensics

Zeek comprehensively logs what it sees and provides a high-level archive of a network's activity.

In-depth Analysis

Zeek comes with analyzers for many protocols, enabling high-level semantic analysis at the application layer.

Highly Stateful

Zeek keeps extensive application-layer state about the network it monitors.

Open Interfaces

Zeek interfaces with other applications for real-time exchange of information.

Open Source

Zeek comes with a BSD license, allowing for free use with virtually no restrictions.

The Zeek Network Security Monitor is not a single tool, more of a powerful network analysis framework (former name Bro)

Source https://www.zeek.org/, redirects to https://www.bro.org/zeek.html

Zeek scripts



source: dns-fire-count.bro from

https://github.com/LiamRandall/bro-scripts/tree/master/fire-scripts https://www.bro.org/sphinx-git/script-reference.html





Now lets do the exercise

Bonus: Zeek on the web 10min

which is number 7 in the exercise PDF.

Exercise setup



We will use a combination of your virtual servers, my switch hardware and my virtual systems.

There will be sniffing done on traffic! Don't abuse information gathered

We try to mimic what you would do in your own networks during the exercises.

Get Started with Zeek



To run in "base" mode: bro -r traffic.pcap

To run in a "near broctl" mode: bro -r traffic.pcap local

To add extra scripts: bro -r traffic.pcap myscript.bro

Note: the project was renamed from Bro to Zeek in Oct 2018





Now lets do the exercise

Bonus: Zeek DNS capturing domain names 10min

which is number 8 in the exercise PDF.





Now lets do the exercise

Bonus: Zeek TLS capturing certificates 10min

which is number 9 in the exercise PDF.

DNS is important



Another tool that provides a basic SQL-frontend to PCAP-files

https://www.dns-oarc.net/tools/packetq

https://github.com/DNS-OARC/PacketQ

Going back in time and finding systems that visited a specific domain can explain when and where an infection started.

Deciding on which tool to use, Zeek or PacketQ depends on the situation.

Suricata IDS/IPS/NSM





Suricata is a high performance Network IDS, IPS and Network Security Monitoring engine. Open Source and owned by a community run non-profit foundation, the Open Information Security Foundation (OISF). Suricata is developed by the OISF and its supporting vendors.

http://suricata-ids.org/ http://openinfosecfoundation.org





Now lets do the exercise

Check your Debian VM 10 min

which is number 10 in the exercise PDF.





Now lets do the exercise

VLANs, Routing and RPF - 2h

which is number 11 in the exercise PDF.





Now lets do the exercise

Configuration of DHCP server 30min

which is number 12 in the exercise PDF.





Now lets do the exercise

Bonus: Configuration of Ubound DNS server 20min

which is number 13 in the exercise PDF.





Now lets do the exercise

Bonus: Configuration of BIRD BGP daemon 40min

which is number 14 in the exercise PDF.





Now lets do the exercise

How to Configure Mirror Port 10min

which is number 15 in the exercise PDF.





Now lets do the exercise

Learn about port security - 10 min

which is number 16 in the exercise PDF.

Summary



We started from a basic Ubuntu/Debian server, and we now know more about our network.

We know it is possible to create dashboards and visualizing the data.

What are the next steps?

Questions?





Henrik Lund Kramshøj hlk@zencurity.com @kramse 🗷 🕽 Need help with infrastructure security or pentesting, ask me!

You are always welcome to send me questions later via email