Bachelorabschlussvortrag

Simulative Analyse von Cyber-Angriffen am Beispiel des Netzes der Fakultät für Informatik der TU Dortmund

---

Nils Dunker

8. Februar 2021

Bachelorabschlussvortrag am Lehrstuhl 4 - Informatik

Themeneinführung

Thema

Titel

Simulative Analyse von Cyber-Angriffen am Beispiel des Netzes der Fakultät für Informatik der TU Dortmund

Bild aus Urheberrechtsgründen entfernt

Thema

Titel

Simulative Analyse von Cyber-Angriffen am Beispiel des Netzes der Fakultät für Informatik der TU Dortmund

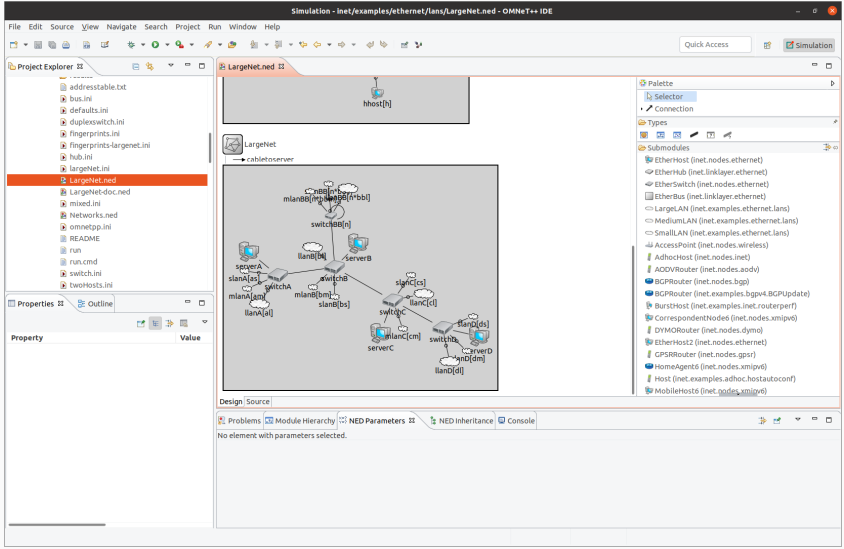
Bild aus Urheberrechtsgründen entfernt

- Drei Angriffe
  - ARP-Spoofing
  - SYN-Flooding
  - Port-Scanner
- Fokus auf Angriffssimulator
- Weitere Einschränkungen
  - Nicht Programmieren
  - Arbeit im direkten Programmökosystem

Themeneinführung

OMNeT++, INET und SEA++

OMNeT++ und INET



SEA++

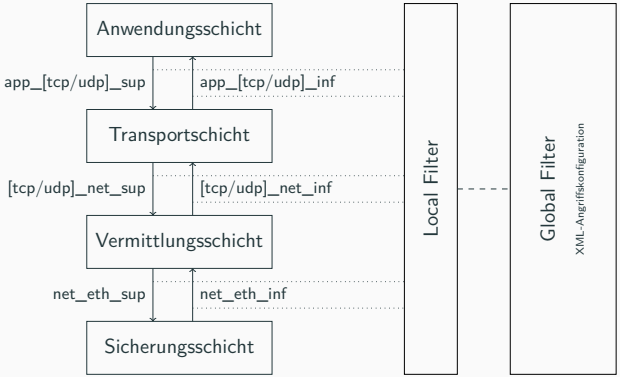

```
1   list dstList = {6}
2
3   from 10 every 1 do {
4   packet examplePacket
5
6   create(examplePacket, "APP.type", "1001")
7
8   change(examplePacket, "APP.info", 123)
9   change(examplePacket, "APP.name", "examplePacket")
10  change(examplePacket, "controlInfo.destAddr", "10.0.0.5")
11  change(examplePacket, "controlInfo.sockId", 4)
12  change(examplePacket, "controlInfo.interfaceId", 0)
13  change(examplePacket, "controlInfo.destPort", 333)
14  change(examplePacket, "sending.outputGate", "app_udp_inf$o[0]")
15
16  put(examplePacket, dstList, TX, FALSE, 0)
17  }
```


```
1  <?xml version="1.0"?>
2  <configuration>
3    <Physical>
4      <Attack>
5        <start_time>1</start_time>
6        <node>9</node>
7        <action>
8          <name>Disable</name>
9        </action>
10     </Attack>
11   </Physical>
12 </configuration>
```

DSL
ASL/ADL

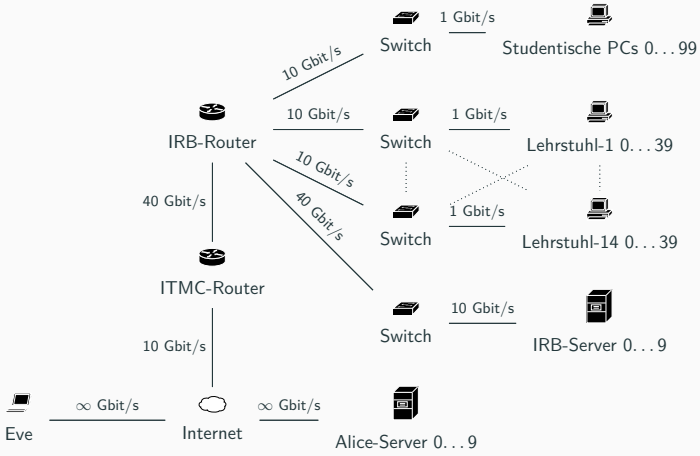
Interpreter
ASI

Engine
ASE

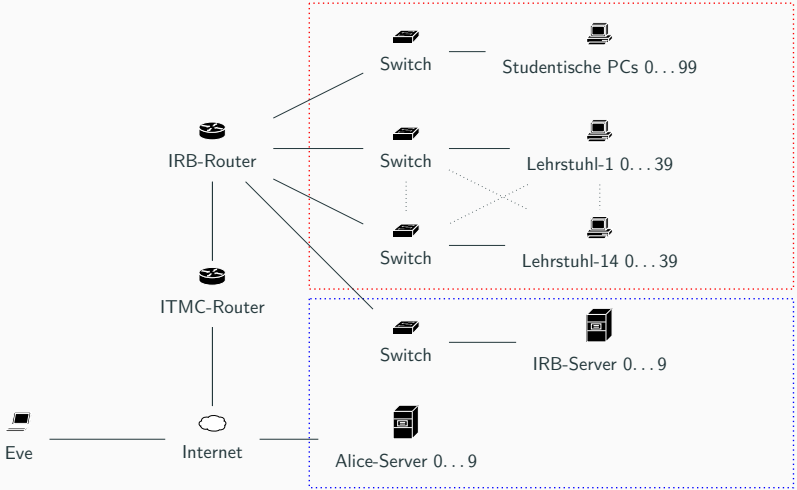


Netzwerkmodellierung

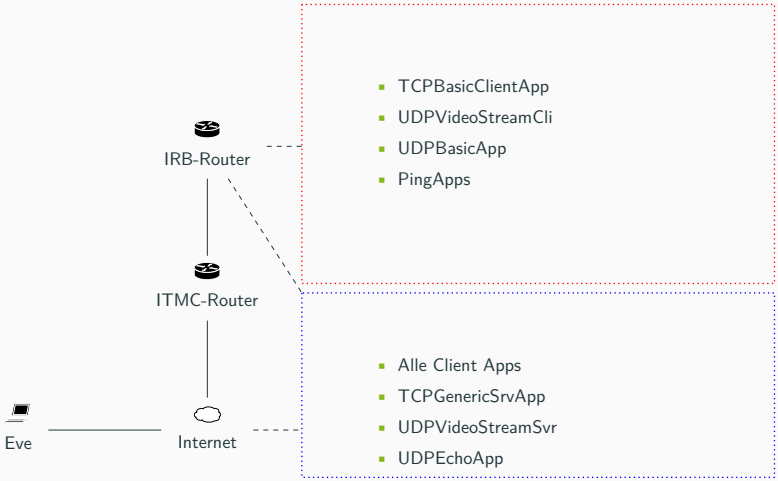
Netzwerkmodell



Netzwerkmodell



Netzwerkmodell



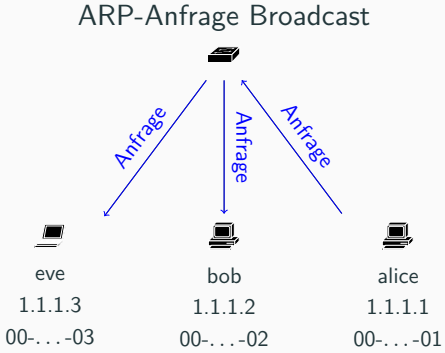
Client Bereich

Server Bereich

Netzwerkmodellierung

ARP-Spoofing

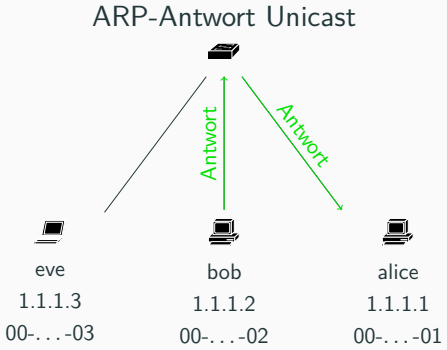
ARP-Spoofing Ablauf



Anfrage (alice⇒alle)

Feld	Wert
SHA	00-00-00-00-00-01
SPA	1.1.1.1
THA	FF-FF-FF-FF-FF-FF
TPA	1.1.1.2

ARP-Spoofing Ablauf

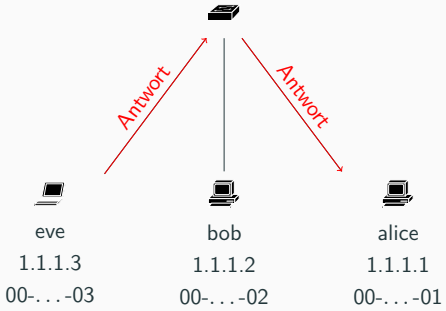


Antwort (bob⇒alice)

Feld	Wert
SHA	00-00-00-00-00-02
SPA	1.1.1.2
THA	00-00-00-00-00-01
TPA	1.1.1.1

ARP-Spoofing Ablauf

ARP-Spoofed-Antwort Unicast



Antwort (eve⇒alice)

Feld	Wert
SHA	00-00-00-00-00-03
SPA	1.1.1.2
THA	00-00-00-00-00-01
TPA	1.1.1.1

ARP-Spoofing Umsetzung

Probleme

- fehlende Hilfsfunktionen
- Unterscheidung von Paketfeldern
- falsche filter-Implementierung

Datenpakete filtern

```
1  list dstList = {3}
2
3  from 0 nodes in dstList do {
4      filter("MAC.opcode" == 1 and "MAC.srcMACAddress" != "0A-AA-00-00-00-0C")
```

ARP-Spoofing Umsetzung

Probleme

- Vorinitialisierung

Daten auslesen

```
8       var srcMac = "Empty"
9       var srcIp = "Empty"
10      var dstIp = "Empty"
11
12      retrieve(original, "MAC.srcMACAddress", srcMac)
13      retrieve(original, "MAC.srcIPAddress", srcIp)
14      retrieve(original, "MAC.destIPAddress", dstIp)
```

ARP-Spoofing Umsetzung

Probleme

- fehlende ARP-Implementierung

Neues ARP-Datenpaket erstellen

```
16      create(arpPacket, "MAC.type", "0040")
17
18      change(arpPacket, "MAC.opcode", 2)
19      change(arpPacket, "MAC.srcMACAddress", "0A-AA-00-00-00-0C" )
20      change(arpPacket, "MAC.destMACAddress", srcMac)
21      change(arpPacket, "MAC.srcIPAddress", dstIp)
22      change(arpPacket, "MAC.destIPAddress", srcIp)
23
24      change(arpPacket, "controlInfo.src", "0A-AA-00-00-00-0C")
25      change(arpPacket, "controlInfo.dest", srcMac)
26      change(arpPacket, "controlInfo.etherType", 2054)
```

ARP-Spoofing Umsetzung

Probleme

- falsche Dokumentation

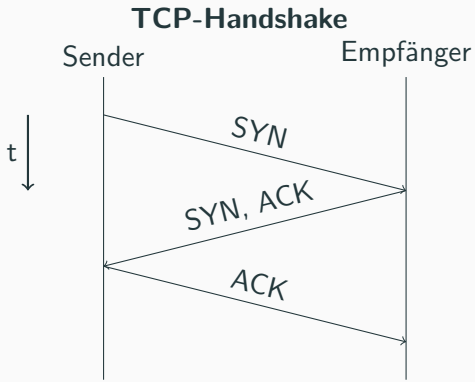
Letzte Änderungen

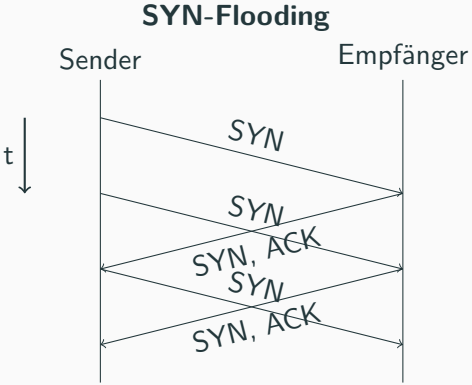
```
28      change(arpPacket, "sending.outputGate", "net_eth_inf$o[0]")
29
30      drop(original, 0)
31
32      send(arpPacket, 0.001)
```

Netzwerkmodellierung

SYN-Flooding

SYN-Flooding

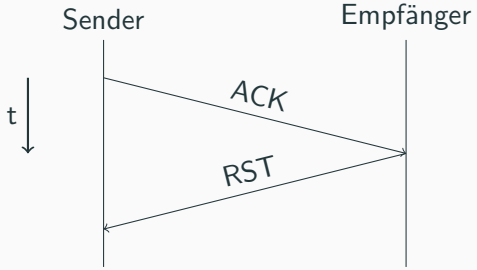




Netzwerkmodellierung

Firewall-Scanner

ACK-Firewall-Scanner

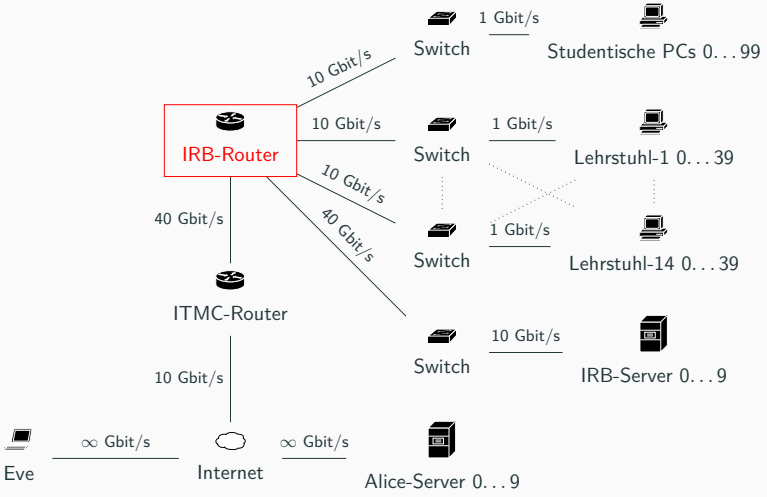


Probleme

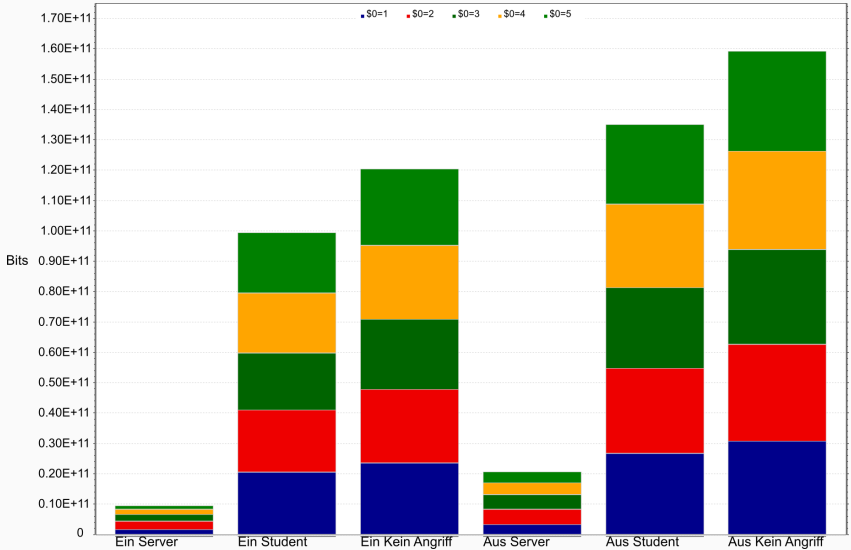
- Wissen über Zustand des Systems
  - Wurde das Gerät bereits getestet?
- Firewall-Module
- Port-Range-Definieren

Ergebnisse

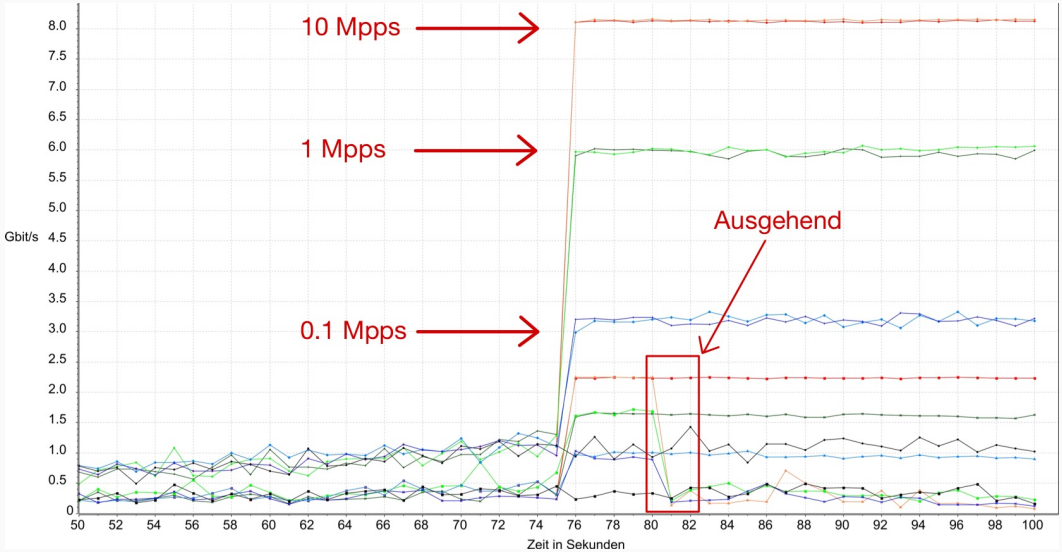
Messpunkt


1 Gbit/s
Switch
Studentische PCs 0...99
10 Gbit/s
10 Gbit/s
IRB-Router
1 Gbit/s
Switch
Lehrstuhl-1 0...39
10 Gbit/s
40 Gbit/s
40 Gbit/s
1 Gbit/s
Switch
Lehrstuhl-14 0...39
ITMC-Router
10 Gbit/s
Switch
IRB-Server 0...9
10 Gbit/s
∞ Gbit/s
∞ Gbit/s
Eve
Internet
Alice-Server 0...9

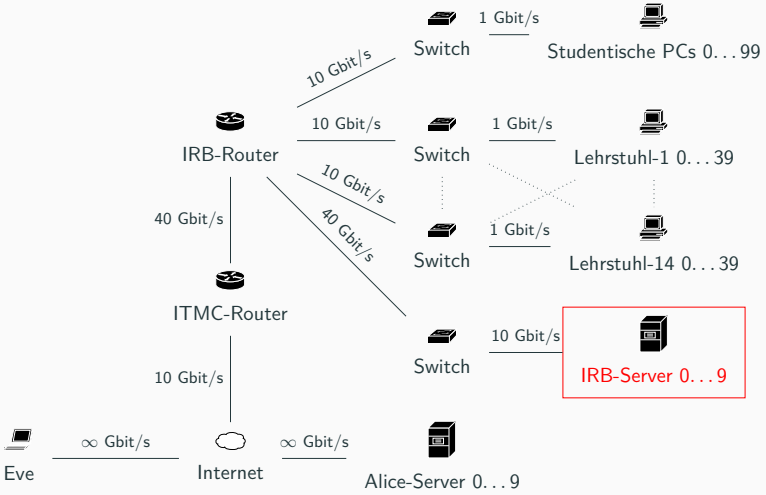
ARP-Spoofing



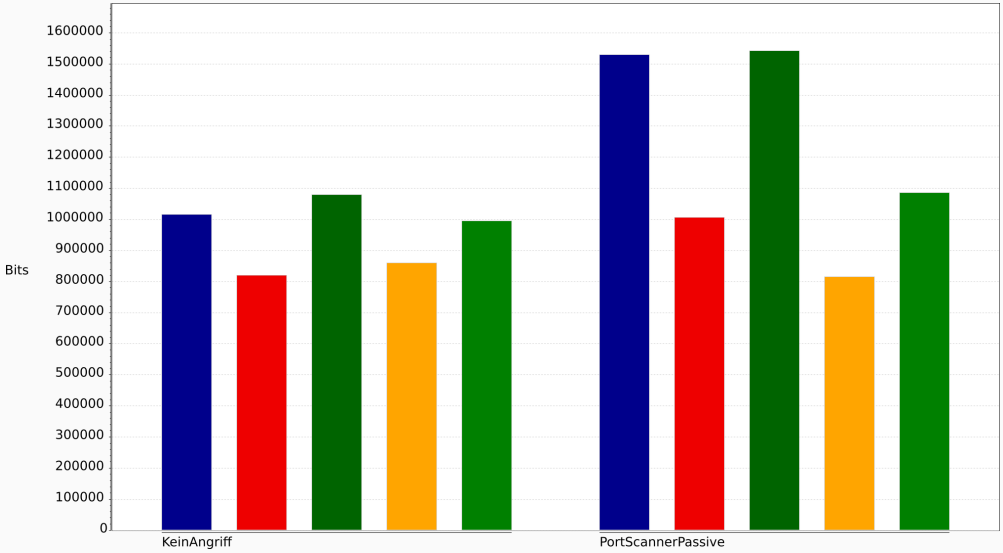
SYN-Flooding



Messpunkt


1 Gbit/s
Switch
Studentische PCs 0...99
10 Gbit/s
10 Gbit/s
IRB-Router
Switch
1 Gbit/s
Lehrstuhl-1 0...39
10 Gbit/s
40 Gbit/s
40 Gbit/s
Switch
1 Gbit/s
Lehrstuhl-14 0...39
ITMC-Router
Switch
10 Gbit/s
IRB-Server 0...9
10 Gbit/s
∞ Gbit/s
∞ Gbit/s
Eve
Internet
Alice-Server 0...9

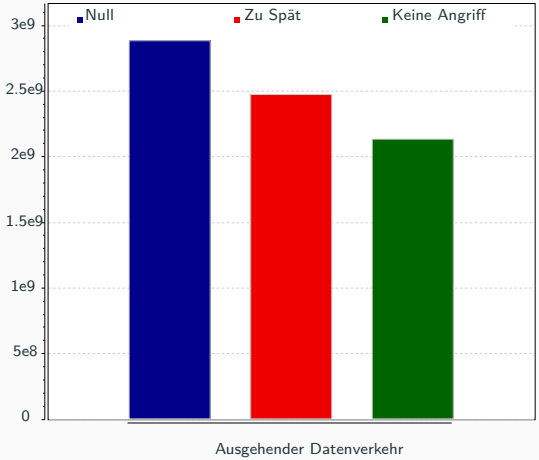
Firewall-Scanner



Zufall in OMNeT++

Implementierung des Zufalls

- Jede Konfiguration besitzt ein Seed
- Seed kann festgelegt werden
- Zufallsgenerator erzeugt Sequenz aus Zahlen
  - ⇒ Immer gleiche Reihenfolge



Zusammenfassung

Einordnung der Probleme

Primäre

- Zufallszahlen
- Dokumentation

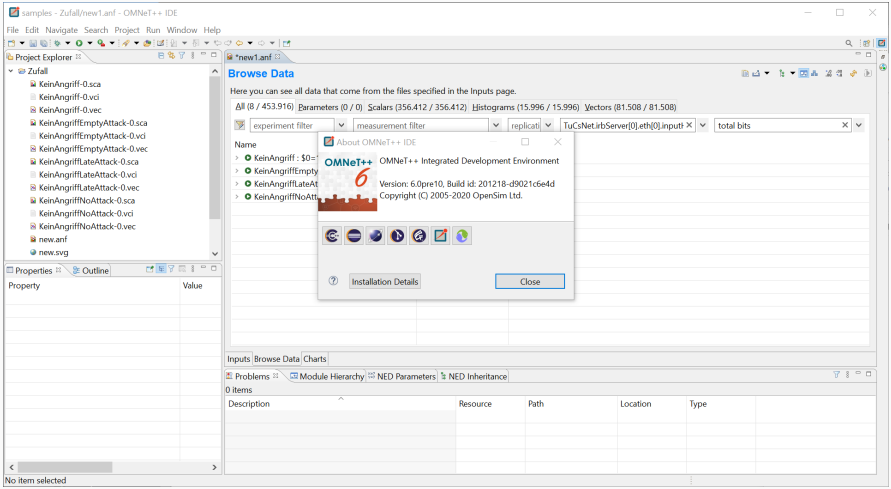
Sekundäre

- Flexibilität
- Modularität
- Erweiterbarkeit
- ADL-Umfang

Weitere

- Veraltete Basis
- Analysetools
- Fehlende Module

Aussicht


samples - Zufall/new1.anf - OMNeT++ IDE
File Edit Navigate Search Project Run Window Help
Project Explorer
Zufall
KeinAngriff-0.sca
KeinAngriff-0.vci
KeinAngriff-0.vec
KeinAngriffEmptyAttack-0.sca
KeinAngriffEmptyAttack-0.vci
KeinAngriffEmptyAttack-0.vec
KeinAngriffLateAttack-0.sca
KeinAngriffLateAttack-0.vci
KeinAngriffLateAttack-0.vec
KeinAngriffNoAttack-0.sca
KeinAngriffNoAttack-0.vci
KeinAngriffNoAttack-0.vec
new.anf
new.svg
Properties
Outline
Property
Value
No item selected
*new1.anf
Browse Data
Here you can see all data that come from the files specified in the Inputs page.
All (8 / 453.916) Parameters (0 / 0) Scalars (356.412 / 356.412) Histograms (15.996 / 15.996) Vectors (81.508 / 81.508)
experiment filter
measurement filter
replicati
TuCsNet.irbServer[0].eth[0].inputti
total bits
Name
KeinAngriff : $0=
KeinAngriffEmpty
KeinAngriffLateAt
KeinAngriffNoAtt
About OMNeT++ IDE
OMNeT++
OMNeT++ Integrated Development Environment
Version: 6.0pre10, Build id: 201218-d9021c6e4d
Copyright (C) 2005-2020 OpenSim Ltd.
Installation Details
Close
Inputs
Browse Data
Charts
Problems
Module Hierarchy
NED Parameters
NED Inheritance
0 items
Description
Resource
Path
Location
Type

Literatur i

[1] *Apache® Subversion® - Ënterprise-class centralized version control for the masses"*. 2020. URL: https://subversion.apache.org/.

[2] Tim Berners-Lee, Roy T. Fielding und Henrik Frystyk Nielsen. *Hypertext Transfer Protocol – HTTP/1.0*. RFC 1945. http://www.rfc-editor.org/rfc/rfc1945.txt. RFC Editor, Mai 1996. URL: http://www.rfc-editor.org/rfc/rfc1945.txt.

[3] *BigBlueButton - Open Source Web Conferencing*. 2020. URL: https://bigbluebutton.org/.

[4] T. Bradley, C. Brown und A. Malis. *Inverse Address Resolution Protocol*. RFC 2390. RFC Editor, Sep. 1998.

Literatur ii

[5] *Cybercrime, Bundeslagebild 2019*. 2019. URL: https://www.bka.de/DE/AktuelleInformationen/StatistikenLagebilder/Lagebilder/Cybercrime/cybercrime_node.html.

[6] *Die Lage der IT-Sicherheit in Deutschland 2019*. 2019. URL: https://www.bsi.bund.de/DE/Publikationen/Lageberichte/lageberichte_node.html.

[7] Simon Yusuf Enoch u. a. "Composite Metrics for Network Security Analysis". In: *Software Networking, 2018(1), 137-160* (7. Juli 2020). arXiv: 2007.03486v2 [cs.CR].

[8] Henrik Frystyk. *The Internet Protocol Stack*. Juli 1994. URL: https://www.w3.org/People/Frystyk/thesis/TcpIp.html.

Literatur iii

[9] Fabien Geyer, Stefan Schneele und Georg Carle. “RENETO, a Realistic Network Traffic Generator for OMNeT++/INET”. In: ICST, Juli 2013. DOI: 10.4108/icst.simutools.2013.251697.

[10] *Git*. 2020. URL: https://git-scm.com/.

[11] *Glossar der Cyber-Sicherheit*. 2020. URL: https://www.bsi.bund.de/DE/Themen/Cyber-Sicherheit/Empfehlungen/cyberglossar/Functions/glossar.html?cms_lv2=9817302.

[12] *GNU Screen*. 2020. URL: https://www.gnu.org/software/screen/.

[13] *Google Public DNS*. 2020. URL: https://developers.google.com/speed/public-dns.

Literatur iv

[14] M. Handley und E. Rescorla and. *Internet Denial-of-Service Considerations*. RFC 4732. RFC Editor, Dez. 2006.

[15] Hubert Hundt. *Cyber-Angriff auf die Ruhr-Universität Bochum*. Ruhr-Universität Bochum. 7. Mai 2020. URL: https://news.rub.de/presseinformationen/servicemeldungen/2020-05-07-digitale-lehre-laeuft-weiter-cyber-angriff-auf-die-ruhr-universitaet-bochum.

[16] "IEEE Standard for Local and Metropolitan Area Network–Bridges and Bridged Networks". In: *IEEE Std 802.1Q-2018 (Revision of IEEE Std 802.1Q-2014)* (Juli 2018), S. 1–1993. DOI: 10.1109/IEEESTD.2018.8403927.

[17] *INET Framework for OMNeT++*. 6. Nov. 2014. URL: https://github.com/inet-framework/inet/tree/v2.6.0.

[18] *INET Framework for OMNeT++/OMNEST*. 2017. URL: https://doc.omnetpp.org/inet/api-old/neddoc/index.html.

[19] *inet-framework/inet*. 2020. URL: https://github.com/inet-framework/inet/releases.

[20] Peter B. Kraft und Andreas Weyert. *Network Hacking: professionelle Angriffs- und Verteidigungstechniken gegen Hacker und Datendiebe ; [Tools für Angriff und Verteidigung - vom Keylogger bis zum Rootkit ; Vorbeugung gegen Malware-Attacken aus dem Internet ; Effektive Schutzmaßnahmen für Privat- und Firmennetze]*. ger. 2., aktualisierte Aufl. Know-how ist blau. OCLC: 642325327. Poing: Franzis, 2010. ISBN: 9783645600309.

[21] OpenSim Limited. *A Quick Overview of the OMNeT++ IDE*. 2020. URL: https://omnetpp.org/documentation/ide-overview/.

[22] OpenSim Limited. *OMNeT++ - Simulation Manual. OMNeT++ version 5.6.1*. URL: https://doc.omnetpp.org/omnetpp/manual/.

[23] OpenSim Limited. *What is OMNeT++?* 2020. URL: https://omnetpp.org/intro/.

[24] OpenSim Limited. "Working with OMNeT++: Bird's-eye view". In: (22. Aug. 2020).

[25] M. Mathis u. a. *TCP Selective Acknowledgment Options*. RFC 2018. RFC Editor, Okt. 1996.

[26] Matthew Monte. *Network Attacks & Exploitation*. John Wiley & Sons, Inc, Feb. 2015. DOI: 10.1002/9781119183440. URL: https://doi.org/10.1002/9781119183440.

[27] T. Narten, E. Nordmark und W. Simpson. *Neighbor Discovery for IP Version 6 (IPv6)*. RFC 2461. RFC Editor, Dez. 1998.

[28] *NETA: A NETwork Attacks Framework*. 2013. URL: https://nesg.ugr.es/index.php/en/neta-2.

[29] *Network Topology Icons*. 2020. URL: https://www.cisco.com/c/en/us/about/brand-center/network-topology-icons.html.

[30] *OMNeT++ Installation Guide Version 4.6*. 2014. URL: https://doc.omnetpp.org/omnetpp4/InstallGuide.pdf.

[31] Jianli Pan und Raj Jain. *A Survey of Network Simulation Tools: Current Status and Future Developments*. 2008. URL: https://www.cse.wustl.edu/~jain/cse567-08/ftp/simtools/index.html.

[32] David C. Plummer. *Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit Ethernet address for transmission on Ethernet hardware*. STD 37. http://www.rfc-editor.org/rfc/rfc826.txt. RFC Editor, Nov. 1982. URL: http://www.rfc-editor.org/rfc/rfc826.txt.

[33] Jon Postel. *Transmission Control Protocol*. STD 7. http://www.rfc-editor.org/rfc/rfc793.txt. RFC Editor, Sep. 1981. URL: http://www.rfc-editor.org/rfc/rfc793.txt.

[34] Stacy Prowell, Michael Borkin und Robert Kraus. *Seven Deadliest Network Attacks*. Mai 2010. DOI: 10.1016/C2009-0-61914-0.

[35] *Python*. 2020. URL: https://www.python.org/about/.

Literatur x

[36] Francesco Racciatti u. a. *SEA++; user manual; SEA++ with SDN support INET-based*. 7. Jan. 2017. URL: https://github.com/seapp/seapp_stable/blob/master/seapp-manual/seapp-manual.pdf.

[37] K. Ramakrishnan, S. Floyd und D. Black. *The Addition of Explicit Congestion Notification (ECN) to IP*. RFC 3168. http://www.rfc-editor.org/rfc/rfc3168.txt. RFC Editor, Sep. 2001. URL: http://www.rfc-editor.org/rfc/rfc3168.txt.

[38] J. Reynolds. *Assigned Numbers: RFC 1700 is Replaced by an On-line Database*. RFC 3232. RFC Editor, Jan. 2002.

[39] J. Reynolds und J. Postel. *Assigned Numbers*. RFC 1700. RFC Editor, Okt. 1994.

[40] Keith W. Ross und James F. Kurose. *Computernetze ein Top-Down-Ansatz mit Schwerpunkt Internet*. 2002.

[41] Theodore John Socolofsky und Claudia Jeanne Kale. *TCP/IP tutorial*. RFC 1180. http://www.rfc-editor.org/rfc/rfc1180.txt. RFC Editor, Jan. 1991. URL: http://www.rfc-editor.org/rfc/rfc1180.txt.

[42] R. Stewart u. a. *Stream Control Transmission Protocol*. RFC 2960. RFC Editor, Okt. 2000.

[43] *tcp(7) - Linux man page*. 2020. URL: https://linux.die.net/man/7/tcp.

[44] *The R Project for Statistical Computing*. 2020. URL: https://www.r-project.org/.

[45] Marco Tiloca, Francesco Racciatti und Gianluca Dini. “Simulative evaluation of security attacks in networked critical infrastructures”. In: *International Conference on Computer Safety, Reliability, and Security*. Springer. 2014, S. 314–323.

[46] Marco Tiloca u. a. “SEA++: A Framework for Evaluating the Impact of Security Attacks in OMNeT++/INET”. In: *Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem*. Hrsg. von Antonio Virdis und Michael Kirsche. Cham: Springer International Publishing, 2019, S. 253–278. ISBN: 978-3-030-12842-5. DOI: 10.1007/978-3-030-12842-5_7. URL: https://doi.org/10.1007/978-3-030-12842-5_7.

[47] J. Touch. *Recommendations on Using Assigned Transport Port Numbers*. BCP 165. RFC Editor, Aug. 2015.

[48] *Web Almanac 2019*. 2019. URL: https://almanac.httparchive.org/en/2019/page-weight.

[49] Omer Yoachimik und Vivek Ganti. *Network-Layer DDoS Attack Trends for Q2 2020*. 5. Aug. 2020. URL: https://blog.cloudflare.com/network-layer-ddos-attack-trends-for-q2-2020/.