

Brucon 2016 Workshop – Start To Pentest ICS

Tijl Deneut Hendrik Derre

Introduction



Hendrik Derre

- Research associate at KU Leuven
- Master's degree in engineering technologies
- Currently active in industrial security research project
- @derre_h / www.linkedin.com/in/hendrikderre

Tijl Deneut

- Researcher and lecturer at Howest University College
- Ethical Hacker
- Currently active in industrial security research project
- @tijldeneut / <u>linkedin.com/in/tijldeneut</u>



Introduction

What to expect?

- ICS Crash course (30 min)
 - Industrial Control systems
 - Programmable Logic controller
 - Industrial communication
- ICS pen testing (30 min)
 - Introducing our environment
 - ICS vulnerabilities
 - Start to pentest ICS

ICS Hacking FUN (Full 2 days)





"An ICS is an broad class of command and control networks and systems that are used to support all types of industrial processes."

They include a variety of system types including:

- supervisory control and data acquisition (SCADA) systems,
- distributed control systems (DCS),
- process control systems (PCS),
- safety control systems (SIS),
- smaller control systems configurations such as programmable logic controllers (PLC's).





Supervision Network

Production Network



7







Where can we find Industrial Control systems?







Programmable Logic Controller



PLC: Programmable Logic Controller

A Solid-state control system that has a user-programmable memory for storing instructions for the purpose of implementing specific functions such as I/O control, logic, timing, counting, three mode (PID) control, communication, arithmetic, and data and file processing.



Feature		CPU 1211C	CPU 1212C	CPU 1214C	CPU 1215C
Physical size (m	m)	90 x 100 x 75	90 x 100 x 75	110 x 100 x 75	130 x 100 x 75
User memory	Work	30 Kbytes	50 Kbytes	75 Kbytes	100 Kbytes
Load		1 Mbyte	1 Mbyte	4 Mbytes	4 Mbytes
	Retentive	10 Kbytes	10 Kbytes	10 Kbytes	10 Kbytes
Local on-board	Digital	6 inputs/4 outputs	8 inputs/6 outputs	14 inputs/10 outputs	14 inputs/10 outputs
I/O	Analog	2 inputs	2 inputs	2 inputs	2 inputs / 2 outputs
Process image	Inputs (I)	1024 bytes	1024 bytes	1024 bytes	1024 bytes
size	Outputs (Q)	1024 bytes	1024 bytes	1024 bytes	1024 bytes
Bit memory (M)		4096 bytes	4096 bytes	8192 bytes	8192 bytes
Signal module (S	SM) expansion	None	2	8	8
Signal board (SE (BB), or commun	 Battery board nication board (CB) 	1	1	1	1
Communication (left-side expans	module (CM) ion)	3	3	3	3
High-speed counters	Total	3 built-in I/O, 5 with SB	4 built-in I/O, 6 with SB	6	6
	Single phase	3 at 100 kHz SB: 2 at 30 kHz	3 at 100 kHz 1 at 30 kHz SB: 2 at 30 kHz	3 at 100 kHz 3 at 30 kHz	3 at 100 kHz 3 at 30 kHz
	Quadrature phase	3 at 80 kHz SB: 2 at 20 kHz	3 at 80 kHz 1 at 20 kHz SB: 2 at 20 kHz	3 at 80 kHz 3 at 20 kHz	3 at 80 kHz 3 at 20 kHz
Pulse outputs1		4	4	4	4
Memory card		SIMATIC Memory	card (optional)		
Real time clock retention time		20 days, typ. / 12 d	ay min. at 40 degree	s C (maintenance-free	Super Capicator)
PROFINET		1 Ethernet commun	nication port		2 Ethernet communication ports
Real math execu	ution speed	2.3 µs/instruction			
Boolean executi	on speed	0.08 µs/instruction			



eature		CPU 1	211C	CPU 1212C	CPU 1214C
nysical size (m	im)	90 x 1	00 x 75	90 x 100 x 75	110 x 100 x 75
ser memory	Work	30 Kb	ytes	50 Kbytes	75 Kbytes
	Load	1 Mby	te	1 Mbyte	4 Mbytes
	Retentive	10 Kb	ytes	10 Kbytes	10 Kbytes
size	Outputs (Q)	1024 bytes	1024 bytes	1024 bytes	1024 bytes
Bit memory (M)		4096 bytes	4096 bytes	8192 bytes	8192 bytes
Signal module (SM) expansion	None	2	8	8
Signal board (SE (BB), or commu	B), Battery board nication board (CB)	1	1	1	1
Communication (left-side expans	module (CM) sion)	3	3	3	3
High-speed counters	Total	3 built-in I/O, 5 with SB	4 built-in I/O, 6 with SB	6	6
	Single phase	3 at 100 kHz SB: 2 at 30 kHz	3 at 100 kHz 1 at 30 kHz SB: 2 at 30 kHz	3 at 100 kHz 3 at 30 kHz	3 at 100 kHz 3 at 30 kHz
	Quadrature phase	3 at 80 kHz SB: 2 at 20 kHz	3 at 80 kHz 1 at 20 kHz SB: 2 at 20 kHz	3 at 80 kHz 3 at 20 kHz	3 at 80 kHz 3 at 20 kHz
eal math exec	ution speed	2.3 µs	/instruction		
oolean executi	on speed	0.08 µ	s/instruction		
PROFINET		1 Ethernet commu	unication port		2 Ethernet
Dealer in					communication ports
Real math execu	ution speed	2.3 µs/instruction			
Boolean executi	on speed	0.08 µs/instruction	1		



eature		CPU 1	U 1211C CPU 1212C		CPU 1214C
hysical size (n	nm)	90 x 1	00 x 75	90 x 100 x 75	110 x 100 x 75
lser memory	Work	30 Kb	ytes	50 Kbytes	75 Kbytes
	Load	1 Mby	te	1 Mbyte	4 Mbytes
	Retentive	10 Kb	ytes	10 Kbytes	10 Kbytes
size	Outputs (Q)	1024 bytes	1024 bytes	1024 bytes	1024 bytes
Bit memory (M)		4096 bytes	4096 bytes	8192 bytes	8192 bytes
Signal module	(SM) expansion	None	2	8	8
Signal board (S (BB), or commu	B), Battery board unication board (CB)	1	1	1	1
Communication (left-side expan	n module (CM) ision)	3	3	3	3
High-speed counters	Total	3 built-in I/O, 5 with SB	4 built-in I/O, 6 with SB	6	6
	Single phase	3 at 100 kHz SB: 2 at 30 kHz	3 at 100 kHz 1 at 30 kHz	3 at 100 kHz 3 at 30 kHz	3 at 100 kHz 3 at 30 kHz
			SB: 2 at 30 kHz		
	Quadrature phase	3 at 80 kHz SB: 2 at 20 kHz	3 at 80 kHz 1 at 20 kHz	3 at 80 kHz 3 at 20 kHz	3 at 80 kHz 3 at 20 kHz
			SB: 2 at 20 kHz		
eal math exec	cution speed	2.3 µs	/instruction		
oolean execut	tion speed	0.08 µ	s/instruction		
PROFINET		1 Ethernet commu	unication port		-2 Ethernet
Real math exec	cution speed	2.3 µs/instruction			contributication ports
Boolean execut	tion speed	0.08.us/instruction	1		

10 Megabyte Hard Disk \$3,495*



5440-12 Top Load Drive * Factory rebuilt 10MB cartridge disk drive only A new Cameo Data Systems controller is available for \$1,495 \$4,495 for a brand new Ampes 10MB drive only





MPUTER COMPONENTS

5848 Sepulveda Boulevard Van Nuys, California 91411 213+786-7411

Croth 279 on inquiry cast. 2010. Separatia Boolecard. Man Nuya, California 91411. 213+786-7411

We are the CP/M** and MP/M** specialist of Southern California. We can supply you with the latest CP/M (\$150) or MP/M (\$300 and with Standard BIOS (\$150) or Custom BIOS (\$300). Immediate delivery worldwide. Domestic and foreign inquiries invited...dealers too. **CP Man MP/Mar. Tinderstool Digota Research

Circle 279 on inquiry card.

We are a full service computer retailer. We totally integrate hardware and software into high quality, high reliability systems. Systems for use in development, process control and general business. Word processing naturally, multi tasking and multi processing too.

BYTE bily 1980 291

Why do we even use PLC's?



Key PLC features:

- Reliable & Robust
- Extreme long product lifecycle
- ► Flexible
- Standardised I/O signals
- Easy programming and debugging





All Shapes and Sizes















How it started:

- Serial Communications
- Proprietary Protocols
- Limited Interconnections
- No external Communications





Present:

- Evolving towards Ethernet based communications
- Open Protocols
- Interconnected with other business systems
- Internet connections for remote access









Ethernet Based Protocols

Universal ICS Protocols

- Modbus TCP: TCP/502
- OPC UA: TCP/4840
- OP UA XML: TCP/80, TCP/443

Process automation specific protocols

- EtherCat: UDP/34980
- Ethernet/IP: TCP/44818, UDP/2222,44818
- FL-net: UDP/55000 to 55003
- Fieldbus HSE: TCP/1089-1091, UDP/1089-1091
- HART-IP: TCP/5094, UDP/5094
- PROFINET: TCP/34962-34964, UDP/ 34962-34964
- S7-Comm: TCP/102

Building automation Specific protocols

- BACnet/IP: UDP/47808
- LonTalk: UDP/1628, UDP/1629
- FOX (Tridium/niagara): TCP/1911

Energy Sector specific protocols

- DNP3: TCP/20000, UDP/ 20000
- DLMS/COSEM: TCP/4059, UDP/4059
- ICCP: TCP/102
- IEC 104: TCP/102
- IEE C37.118: TCP/4712, UDP/4713
- MMS: TCP/102

Modbus

- Designed in 1979 by Modicon (now Schneider Electric)
- Widely adopted protocol used in multiple industries
- Mostly used on field level (sometimes between PLC & HMI)
- Open Standard, freely distributed by the modbus organization





Modbus

- Master / slave (Standard 1 master and max 247 slaves in a network)
- Request / reply (Only master can initialize communication)
- Each Modbus device is assigned an unique (in the network) address.
- Function codes used to described the action desired from device

Function Code	Action	Table Name
01 (01 hex)	Read	Discrete Output Coils
05 (05 hex)	Write single	Discrete Output Coil
15 (0F hex)	Write multiple	Discrete Output Coils
02 (02 hex)	Read	Discrete Input Contacts
04 (04 hex)	Read	Analog Input Registers
03 (03 hex)	Read	Analog Output Holding Registers
06 (06 hex)	Write single	Analog Output Holding Register
16 (10 hex)	Write multiple	Analog Output Holding Registers



Modbus

- Lack of Authentication
- Lack of Encryption

Lack of broadcast suppression

153 0.206422	141.81.0.10	141.81.0.66	Modbus/TCP	66 Query: Trans:	1426; Unit: 255, Func:	2: Read Discrete Inputs
168 0.248615	141.81.0.46	141.81.0.10	Modbus/TCP	78 Response: Trans: 2	28215; Unit: 255, Func:	15: Write Multiple Coils
169 0.249547	141.81.0.26	141.81.0.10	Modbus/TCP	340 Response: Trans: 3	18527; Unit: 255, Func:	4: Read Input Registers

> Frame 153: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)

> Ethernet II, Src: HewlettP_e0:02:5e (78:e7:d1:e0:02:5e), Dst: ElauAg_02:36:35 (00:04:17:02:36:35)

> Internet Protocol Version 4, Src: 141.81.0.10, Dst: 141.81.0.66

> Transmission Control Protocol, Src Port: 54138 (54138), Dst Port: 502 (502), Seq: 13, Ack: 12, Len: 12

Modbus/TCP

Transaction Identifier: 1426 Protocol Identifier: 0

Length: 6

Unit Identifier: 255

Modbus

Function Code: Read Discrete Inputs (2) Reference Number: 0 Bit Count: 11



SIEMENS 57-Comm

S7 Protocol, is the backbone of the Siemens communications,

Ethernet implementation relies on ISO TCP (RFC1006)

S7 Protocol is Function oriented or Command oriented



SIEMENS *57-Comm*

- Each command consists of
 - A header
 - A set of parameters
 - A parameters data (optional)
 - A data block (optional)



SIEMENS *57-Comm*

- Each command consists of
 - A header
 - A set of parameters
 - A parameters data (optional)
 - A data block (optional)

	CPU					
	300	400	WinAC	Snap7S	1200	1500
DB Read/Write	0	0	0	0	0	O(3)
EB Read/Write	0	0	0	0	0	0
AB Read/Write	0	0	0	0	0	0
MK Read/Write	0	0	0	0	0	0
TM Read/Write	0	0	0	0	-	-
CT Read/Write	0	0	0	0	-	-
Read SZL	0	0	0	0	0	0
Multi Read/Write	0	0	0	0	0	0
Directory	0	0	0	0	-	-
Date and Time	0	0	0	0	-	-
Control Run/Stop	0	0	0	0	-	-
Security	0	0	0	0	-	-
Block Upload/Down/Delete	0	0	0	-	-	-

SIEMENS 57-Comm

Lack of Authentication

Lack of Encryption

1	23 3.498105	192.168.1.40	192.168.1.10	S7COMM	123 ROSCTR:[Userdata] Function:[Response] -> [CPU functions] -> [Read SZL] ID=0x0111 Index=0x
1	25 3.498548	192.168.1.10	192.168.1.40	S7COMM	83 ROSCTR:[Userdata] Function:[Request] -> [Time functions] -> [Read clock]
	26 3.503074	192.168.1.40	192.168.1.10	S7COMM	97 ROSCTR:[Userdata] Function:[Response] -> [Time functions] -> [Read clock]
	28 3.554092	192.168.1.10	192.168.1.40	S7COMM	87 ROSCTR:[Userdata] Function:[Request] -> [CPU functions] -> [Read SZL] ID=0x0132 Index=0x0
	29 3 559093	192 168 1 40	192 168 1 10	SZCOMM	135 ROSCTR · [Userdata] Function · [Response] -> [CPU functions] -> [Read S71] TD=0x0132 Index=0x

> Frame 26: 97 bytes on wire (776 bits), 97 bytes captured (776 bits)

- Ethernet II, Src: Siemens_23:eb:3b (00:1b:1b:23:eb:3b), Dst: AsustekC_84:5e:41 (90:e6:ba:84:5e:41)
- > Internet Protocol Version 4, Src: 192.168.1.40, Dst: 192.168.1.10
- > Transmission Control Protocol, Src Port: 102 (102), Dst Port: 4173 (4173), Seq: 498, Ack: 244, Len: 43

> TPKT, Version: 3, Length: 43

- > ISO 8073/X.224 COTP Connection-Oriented Transport Protocol
- ✓ S7 Communication
 - > Header: (Userdata)
 - > Parameter: (Response) ->(Time functions) ->(Read clock)
 - Data: (Timestamp: Aug 20, 2014 11:59:43.912)
 Return code: Success (0xff)
 - Transport size: OCTET STRING (0x09)

Length: 10

> S7 Timestamp: Aug 20, 2014 11:59:43.912



Remote access over internet to ICS networks

what can go wrong?



Remote access over internet to ICS networks

what can go wrong?



Remote access over internet to ICS networks

what can go wrong?

Shodan Deve	lopers Book View All			Show AP	l Ke
🔏 Shodan	siemens simatic	Q Explore	Downloads Reports Enterprise Access Contac	t Us 🕹 My Account Upg	gra
🔏 Exploits 🛛 🔏	Maps Share Search	📥 Download Results 🛛 🔟 Create Repo	rt		
		Total results: 904 193.95.106.249 Agence Tunisienne Internet - ATI Added on 2016-10-25 11:21:10 GMT Tunisia Details	Siemens, SIMATIC, S7-200		
Taiwan, Province of Cl Tunisia Germany Italy United States TOP SERVICES Siemens S7 SNMP Modbus	hina 147 125 71 57 49 430 321 95	175-96-80-224. dynamic.tfn.ent.tw Taiwan Mobile Co., Ltd. Added on 2016-10-25 10:00:51 GMT	Location designation of a module: Copyright: Original Siemens Equipment Module type: IM151-8 PN/DP CPU PLC name: outlet Module: v.0.0 Plant identification: Power Corporation OEM ID of a module: Module name: Siemens, SIMATIC, S7-300 Serial number of module: 16111663		
60450 60402 TOP ORGANIZATIONS Taiwan Mobile Co., Ltd Tunisia BackBone TOPNET Taiwan Fixed Network, Deutsche Telekom AG	1 1 5 1 5 5 1 5 5 5 5 8 5 5 8 5 7 8 5 5 8 5 7 8 26	31.7.243.227 WIA spol. s.r.o. Added on 2018-10-25 09:52:48 GMT Czech Republic, Prague Details	Copyright: Original Siemens Equipment PLC name: Stazione SIMATIC 300 Module type: CPU 315-2 DP Unknown (129): Boot Loader A% Module: 6ES7 315-2AH14-0AB0 v.0.5 Basic Firmware: v.3.3.10 Module name: CPU 315-2 DP Serial number of module: S C-E3T885952014 Plant identification: Basic Hardw		

TOP OPERATING SYSTEMS

Protocols

BACnet: 10,530 DNP3: 588 EtherNet/IP: 3,943 Modbus: 13,949 Niagara Fox: 23,294 Niagara Fox with SSL: 159 Siemens S7: 2,701

SHODANICS Radar

About

The Shodan search engine has started to crawl the Internet for protocols that provide raw, direct access to industrial control systems (ICS). This visualization shows the location of these industrial control systems on the Internet as well as other related data.

REYKJAVIK MSTERDAM ٩ BUCHAREST 911 110 Modbus TCP: 13 949 57comm: 2 701



Contact

For all inquiries relating to Shodan or the ICS Radar please contact:

support@shodan.io Twitter: @shodanhq

Share Tweet 😚 🛖 🔶 7 points

Remote access over internet to ICS networks

what can go wrong?



Industrial Research

ICS Vulnerabilities



General Vulnerability Overview

In **BELGIUM** (focus for my research), there are three most-used PLC vendors:







Early vulnerabilities

- Beckhoff: http://<ip>/config has a website with a Session ID vulnerability
 - CVE-2014-4051, disclosed in 2015, fixed 3.1.4018.13 (August 2015)
 - Authentication bypass, exploit: <u>https://www.exploit-db.com/exploits/38514</u>
 - PLUS a lot of issues with the used Windows CE (6), CERDisplay anyone?

- Siemens:
 - CVE-2013-5709, Scalance Switches have a website with a Session ID vulnerability, disclosed by Eireann Leverett (blackswanburst) fixed with firmware **v5.0.0**
 - CVE-2016-6204, XSS vulnerability in newest Scalance Router (Cinema Remote Connect), disclosed this summer by me ⁽²⁾ fixed with v1.2

But things are WORSE

As seen earlier; industrial devices rely on old and insecure protocols.

So some personal research to investigate these protocols ...

- Phoenix Contact: completely proprietary, not even Wireshark has any idea what we are dealing with

Filter: Expression Clear Apply Save No. Time Source Destination Protocol Length Info 1 0.00000 172.20.0.56 172.20.3.10 TCP 66 49261-20547 [SYN] Seq=0 Wit 2 0.001274 172.20.0.56 172.20.3.10 TCP 62 20547-49261 [SYN] AcK Seq=1 AcK 3 0.001992 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, AcK] Seq=1 AcK Seq Seq Seq AcK Seq AcK Seq Seq Seq AcK Seq Seq AcK Seq AcK Seq AcK Seq AcK	Filter: No. Time 1 0.000000 2 0.001274 3 0.001992 4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 C Frame 7: 210 H E thernet II, 5 E Internet Proto	Source 172.20.0.56 172.20.3.10 2 172.20.0.56 4 172.20.3.10 2 172.20.3.10 2 172.20.3.10 2 172.20.3.10 3 172.20.3.10	Destination 172.20.3.10 172.20.3.10 172.20.3.10 172.20.3.10 172.20.0.56 172.20.3.10	Expression C Protocol TCP TCP TCP TCP TCP TCP	Lear Apply Save Length Info 66 49261-20 62 20547-49 54 49261-20 64 49261-20 68 20547-49	78 144 1547 [SYN] Seq=0 wir 1261 [SYN, ACK] Seq= 1547 [ACK] Seq=1 ACL 1547 [PSH, ACK] Seq=
Filter: ✓ Expression Clear Apply Save No. Time Source Destination Protocol Length Info 1 0.000000 172.20.0.56 172.20.3.10 TCP 66 49261-20547 [SYN] Seq=0 win 2 0.001274 172.20.3.10 172.20.3.10 TCP 62 20547-49261 [SYN] AcK] Seq=1 AcK Seq	Filter: No. Time 1 0.00000 2 0.001274 3 0.001992 4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	Source 172.20.0.56 172.20.3.10 2 172.20.0.56 4 172.20.0.56 0 172.20.3.10 2 172.20.0.56 3 172.20.3.10 172.20.3.10	Destination 172.20.3.10 172.20.0.56 172.20.3.10 172.20.3.10 172.20.0.56 172.20.3.10	Expression C Protocol TCP TCP TCP TCP TCP TCP TCP TCP TCP TCP	Iear Apply Save Length Info 66 49261-20 62 20547-49 54 49261-20 64 49261-20 64 49261-20 64 49261-20 64 49261-20 68 20547-49)547 [SYN] Seq=0 wii 1261 [SYN, ACK] Seq= 1547 [ACK] Seq=1 Ack 1547 [PSH, ACK] seq=
No. Time Source Destination Protocol Length Info 1 0.000000 172. 20. 0. 56 172. 20. 3.10 TCP 66 49261-20547 [SYN] Seq=0 Wit 2 0.001274 172. 20. 0. 56 172. 20. 3.10 TCP 62 20547-49261 [SYN, ACK] Seq=1 AcK 4 1.444054 172. 20. 0. 56 172. 20. 3.10 TCP 64 49261-20547 [AKK] Seq=1 AcK 5 1.449140 172. 20. 0. 56 172. 20. 3.10 TCP 64 49261-20547 [PSH, ACK] Seq= 6 1.449392 172. 20. 0. 56 172. 20. 3.10 TCP 64 49261-20547 [PSH, ACK] Seq= 7 1.471269 172. 20. 3.10 172. 20. 3.10 TCP 92 49261-20547 [PSH, ACK] Seq= 9 1.478762 172. 20. 0.56 172. 20. 3.10 TCP 92 49261-20547 [PSH, ACK] Seq= 10 1.478024 172. 20. 0.56 172. 20. 3.10 TCP 92 49261-20547 [PSH, ACK] Seq=	No. Time 1 0.00000 2 0.001274 3 0.001992 4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	Source 0 172.20.0.56 4 172.20.3.10 2 172.20.0.56 4 172.20.3.10 2 172.20.3.10 2 172.20.3.10 2 172.20.3.10 4 172.20.3.10	Destination 172.20.3.10 172.20.0.56 172.20.3.10 172.20.3.10 172.20.0.56 172.20.3.10	Protocol TCP TCP TCP TCP TCP	Length Info 66 49261-20 62 20547-49 54 49261-20 64 49261-20 68 20547-49)547 [SYN] Seq=0 win)261 [SYN, ACK] Seq=)547 [ACK] Seq=1 ACK)547 [PSH, ACK] Seq
1 0.000000 172.20.0.56 172.20.3.10 TCP 66 49261-20547 [SVN] seq=0 wii 2 0.001992 172.20.0.56 172.20.0.56 TCP 62 20547-49261 [SVN, ACK] seq 3 0.001992 172.20.0.56 172.20.3.10 TCP 54 49261-20547 [ACK] seq=1 ACK 4 1.444054 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 5 1.449140 172.20.3.10 172.20.0.56 TCP 68 20547-49261 [PSH, ACK] seq 6 1.449392 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 7 1.471269 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 9 1.478762 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 10 1.478024 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.3.10 TCP 92 20547-49261 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10<	1 0.00000 2 0.001274 3 0.001992 4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	0 172.20.0.56 4 172.20.3.10 2 172.20.0.56 4 172.20.3.10 2 172.20.3.10 2 172.20.56 3 172.20.3.10 4 172.20.3.10	172.20.3.10 172.20.0.56 172.20.3.10 172.20.3.10 172.20.0.56 172.20.3.10	TCP TCP TCP TCP TCP	66 49261-20 62 20547-49 54 49261-20 64 49261-20 68 20547-49	0547 [SYN] Seq=0 wii 0261 [SYN, ACK] Seq= 0547 [ACK] Seq=1 ACK 0547 [PSH, ACK] Seq=
2 0.001274 172.20.3.10 172.20.0.56 TCP 62 20547-49261 [SVN, ACK] seq 3 0.001992 172.20.0.56 172.20.3.10 TCP 54 49261-20547 [ACK] seq=1 AcK 4 1.444054 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 5 1.449140 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 6 1.449392 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 6 1.449392 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 8 1.471502 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 9 1.478762 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.3.10 TCP 92 0547-49261 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 14 1.493212 172.20.3.10	2 0.001274 3 0.001992 4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 4 1.497934 4 1.497934 4 1.497934 5 1.497934 4 1.497934 5 1.497934 5 1.497934 5 1.497934 6 1.497934 6 1.497934 7 1.47120 8 1.497934 7 1.47120 9 1.478762 9 1.478762 1 1.485723 1 2.486034 1 3.491906 1 4.493212 1 5 1.497934 1 5 1.49794 1 5 1.4974 1 5 1.497	4 172.20.3.10 2 172.20.0.56 4 172.20.0.56 0 172.20.3.10 2 172.20.0.56 3 172.20.3.10 172.20.3.10 4 172.20.3.10	172.20.0.56 172.20.3.10 172.20.3.10 172.20.0.56 172.20.3.10	TCP TCP TCP TCP	62 20547→49 54 49261→20 64 49261→20 68 20547→49	9261 [SYN, ACK] Seq 9547 [ACK] Seq=1 Ack 9547 [PSH, ACK] Seq=
3 0.001992 172.20.0.56 172.20.3.10 TCP 54 49261-20547 [ACK] seq=1 Ack 4 1.444054 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 5 1.449140 172.20.3.10 172.20.0.56 TCP 68 20547-49261 [PSH, ACK] seq 6 1.449392 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 7 1.471269 172.20.3.10 172.20.0.56 TCP 210 20547-49261 [PSH, ACK] seq 9 1.471502 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 9 1.478762 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 10 1.479024 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10 TCP 92 20547-49261 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 14 1.493212 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497034 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497034 172.20.3.10 172.20.3.6 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497034 172.20.3.10 172.20.3.5 TCP 80 20547-49261 [PSH, ACK] seq 16 1.49704 173 100 100 1000 1000 1000 100000 100000000	3 0.001992 4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 4 5 1.497934 4 5 1.497934 4 5 1.497934 5 Ethernet II, 5 8 Internet Proto	2 172.20.0.56 4 172.20.0.56 5 172.20.3.10 2 172.20.0.56 3 172.20.3.10 172.20.3.10	172.20.3.10 172.20.3.10 172.20.0.56 172.20.3.10	TCP TCP TCP	54 49261→20 64 49261→20 68 20547→49)547 [ACK] Seq=1 Ack)547 [PSH, ACK] Seq=
4 1.444054 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 5 1.449140 172.20.3.10 172.20.0.56 TCP 68 20547-49261 [PSH, ACK] seq 6 1.449392 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 7 1.471269 172.20.3.6 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 8 1.471502 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 9 1.478762 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 10 1.479024 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 14 1.493212 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 <td< td=""><td>4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 C E Frame 7: 210 H E Ethernet II, S E Internet Proto</td><td>4 172.20.0.56 0 172.20.3.10 172.20.0.56 9 172.20.3.10 172.20.3.10</td><td>172.20.3.10 172.20.0.56 172.20.3.10</td><td>TCP TCP</td><td>64 49261→20 68 20547→49</td><td>)547 [PSH, ACK] Sed=</td></td<>	4 1.444054 5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 C E Frame 7: 210 H E Ethernet II, S E Internet Proto	4 172.20.0.56 0 172.20.3.10 172.20.0.56 9 172.20.3.10 172.20.3.10	172.20.3.10 172.20.0.56 172.20.3.10	TCP TCP	64 49261→20 68 20547→49)547 [PSH, ACK] Sed=
5 1.449140 172.20.3.10 172.20.0.56 TCP 68 20547-49261 [PSH, ACK] seq 6 1.449392 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 7 1.471269 172.20.3.10 172.20.0.56 TCP 210 20547-49261 [PSH, ACK] seq 8 1.471502 172.20.0.56 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 9 1.478762 172.20.3.10 172.20.0.56 TCP 100 20547-49261 [PSH, ACK] seq 10 1.479024 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.3.66 TCP 92 20547-49261 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 14 1.493212 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10	5 1.449140 6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 C Frame 7: 210 H Ethernet II, S Finternet Proto	0 172.20.3.10 2 172.20.0.56 9 172.20.3.10	172.20.0.56 172.20.3.10	TCP	68 20547→49	your front werd bed
6 1.449392 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 7 1.471269 172.20.3.10 172.20.0.56 TCP 210 20547-49261 [PSH, ACK] seq 8 1.471502 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 9 1.478762 172.20.3.10 172.20.0.56 TCP 100 20547-49261 [PSH, ACK] seq 10 1.479024 172.20.3.10 172.20.0.56 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.0.56 TCP 92 20547-49261 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 14 1.493212 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F02001 172.20.3.10	6 1.449392 7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 C Frame 7: 210 H Ethernet II, S Ethernet II, S	2 172.20.0.56 9 172.20.3.10	172.20.3.10	TCD		261 [PSH, ACK] Seq=
7 1.471269 172.20.3.10 172.20.0.56 TCP 210 20547-49261 [PSH, ACK] seq- 8 1.471502 172.20.0.56 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq- 9 1.478762 172.20.3.10 172.20.0.56 TCP 100 20547-49261 [PSH, ACK] seq- 10 1.479024 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq- 11 1.485723 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq- 12 1.486034 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq- 13 1.491906 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq- 14 1.493212 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 <	7 1.471269 8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 ≤ Ethernet II, S B Internet Proto Transmission (9 172.20.3.10		TCP	64 49261→20)547 [PSH, ACK] Seq=
8 1.471502 172.20.0.56 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 9 1.478762 172.20.3.10 172.20.0.56 TCP 100 20547-49261 [PSH, ACK] seq 10 1.479024 172.20.3.10 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 TCP 92 20547-49261 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 14 1.493212 172.20.3.10 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547	8 1.471502 9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 ✓	170 00 0 50	172.20.0.56	TCP	210 20547→49	261 [PSH, ACK] Seq
9 1.478762 172.20.3.10 172.20.0.56 TCP 100 20547-49261 [PSH, ACK] seq 10 1.479024 172.20.0.56 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.0.56 TCP 92 20547-49261 [PSH, ACK] seq 12 1.486034 172.20.3.10 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 14 1.493212 172.20.3.10 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 Frame 7: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits) on interface 0 E thernet II, Src: Phoenixc_09:21:64 (00:a0:45:09:21:64), Dst: Vmware_e7:ca:1b (00:0c:29:e7:ca:1b) B Internet Protocol Version 4, Src: 172.20.3.10, Dst: 172.20.0.56 B Transmission Control Protocol, Src Port: 20547, Dst Port: 49261, Seq: 15, Ack: 21, Len: 156 15 Data (156 bytes) 0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34 	9 1.478762 10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 Frame 7: 210 H Ethernet II, S Internet Proto Transmission C	2 1/2.20.0.56	172.20.3.10	TCP	92 49261→20)547 [PSH, ACK] Seq=
10 1.479024 172.20.0.56 172.20.3.10 TCP 92 49261-20547 [PSH, ACK] seq 11 1.485723 172.20.3.10 172.20.0.56 TCP 92 20547-49261 [PSH, ACK] seq 12 1.486034 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq 13 1.491906 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 14 1.493212 172.20.3.10 172.20.0.56 TCP 80 49261-20547 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 17 1 10 20 2 10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 17 1 10 20 2 10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 17 1 10 20 2 10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq 17 1 10 20 2 10 172.20.20.56 TCP 80 20547-49261 [PSH, ACK] seq 16 1 F03001 172.20.20.56 TCP 80 20547-49261 [PSH, ACK] seq 17 1 10 20 2 10 172.20.20.56 TCP 80 20547-49261 [PSH, ACK] seq 17 1 10 20 2 10 172.20.20.56 TCP 120547 [PSH, ACK] seq 18 Frame 7: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits) on interface 0 19 Transmission Control Protocol, Src Port: 20547, Dst Port: 49261, Seq: 15, ACK: 21, Len: 156 19 Data (156 bytes) 10 030 7f ec a9 9a 00 00 [c 01 c4 00 00 02 92 00 56 34 (2PF0Con 05 V4.2]	10 1.479024 11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 C Ethernet II, S E Internet Proto Transmission C	2 172.20.3.10	172.20.0.56	TCP	100 20547→49	9261 [PSH, ACK] Seq=
11 1.485723 172.20.3.10 172.20.0.56 TCP 92 20547-49261 [PSH, ACK] seq- 12 1.486034 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq- 13 1.491906 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 14 1.493212 172.20.3.10 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 16 1.72.20.3.10 172.20.3.10 TCP 80 20547-49261 [PSH, ACK] seq- 16 1.72.20.3.10 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 17 173.20.3.10 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 18 Frame 7: 210 bytes on wire (1680 bits), 210 bytes captured (168	11 1.485723 12 1.486034 13 1.491906 14 1.493212 15 1.497934 C Frame 7: 210 H Ethernet II, S E Internet Proto Transmission C	4 172.20.0.56	172.20.3.10	TCP	92 49261→20)547 [PSH, ACK] Seq=
12 1.486034 172.20.0.56 172.20.3.10 TCP 64 49261-20547 [PSH, ACK] seq- 13 1.491906 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 14 1.493212 172.20.0.56 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 16 1 00101 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 17 1001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 18 Frame 7: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits) on interface 0 100111 100111 100111 100111 100111 100111 100111 100111 100111 100111 100111 100111 100111 100111 <td>12 1.486034 13 1.491906 14 1.493212 15 1.497934 Frame 7: 210 H Ethernet II, S Internet Proto</td> <td>3 172.20.3.10</td> <td>172.20.0.56</td> <td>TCP</td> <td>92 20547→49</td> <td>261 [PSH, ACK] Seq=</td>	12 1.486034 13 1.491906 14 1.493212 15 1.497934 Frame 7: 210 H Ethernet II, S Internet Proto	3 172.20.3.10	172.20.0.56	TCP	92 20547→49	261 [PSH, ACK] Seq=
13 1.491906 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 14 1.493212 172.20.0.56 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 16 1 F03001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 17 10 001 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 16 1 F03001 172.20.3.10 172.20.0.56 TCP 61 403C1 20141 Incr Incr 17 10 00 1 Version 4, Src: 172.20.3.10, Dst: 172.20.0.56 Transmission Control Protocol, Src Port: 20547, Dst Port: 49261, Seq: 15, Ack: 21, Len: 156 Incr Incr <td>13 1.491906 14 1.493212 15 1.497934 Frame 7: 210 H Ethernet II, S Internet Proto</td> <td>4 172.20.0.56</td> <td>172.20.3.10</td> <td>TCP</td> <td>64 49261→20</td> <td>)547 [PSH, ACK] Seq=</td>	13 1.491906 14 1.493212 15 1.497934 Frame 7: 210 H Ethernet II, S Internet Proto	4 172.20.0.56	172.20.3.10	TCP	64 49261→20)547 [PSH, ACK] Seq=
14 1.493212 172.20.0.56 172.20.3.10 TCP 80 49261-20547 [PSH, ACK] seq- 15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 80 20547-49261 [PSH, ACK] seq- 16 1 10011 172 20.0.56 If Frame 7: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits) on interface 0 If to 10011 172 20.0.56 If to 2011 2011 10011	14 1.493212 15 1.497934 Frame 7: 210 H Ethernet II, S Internet Proto Transmission (5 172.20.3.10	172.20.0.56	TCP	80 20547→49	261 [PSH, ACK] Seq=
15 1.497934 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 16 1 172.20.3.10 172.20.0.56 TCP 80 20547-49261 [PSH, ACK] seq- 17 1 172.20.3.10 172.20.0.56 TCP 61 4024 2014 2014 18 Frame 7: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits) on interface 0 61 4024 2014 2014 2014 19 Ethernet II, Src: Phoenixc_09:21:64 (00:a0:45:09:21:64), Dst: Vmware_e7:ca:1b (00:00:29:e7:ca:1b) 100:00:00:29:e7:ca:1b) 110:00:00:29:e7:ca:1b) 19 Internet Protocol Version 4, Src: 172.20.3.10, Dst: 172.20.0.56 172.20.0.56 110:00:00:00:00:00:00:00:00:00:00:00:00:	15 1.497934	2 172.20.0.56	172.20.3.10	TCP	80 49261→20)547 [PSH, ACK] Seq=
<pre></pre>	 ✓ Frame 7: 210 b ✓ Ethernet II, s ✓ Internet Proto ✓ Transmission (4 172.20.3.10	172.20.0.56	TCP	80 20547→49	9261 [PSH, ACK] Seq=
B Frame 7: 210 bytes on wire (1680 bits), 210 bytes captured (1680 bits) on interface 0 B Ethernet II, Src: Phoenixc_09:21:64 (00:a0:45:09:21:64), Dst: Vmware_e7:ca:1b (00:0c:29:e7:ca:1b) B Internet Protocol Version 4, Src: 172.20.3.10, Dst: 172.20.0.56 B Transmission Control Protocol, Src Port: 20547, Dst Port: 49261, Seq: 15, Ack: 21, Len: 156 B Data (156 bytes) 0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34 20 32 20 56 34 20 32 20 56 34 20 32 20 56 34 20 32 20 56 34 20 32 20 56 34 20 32 20 54 4.2 50 72 6f 43 6f 6e 4f 53 20 56 34 20 32 20 56 34 20 32 20 54 5.2 50 72 6f 43 6f 6e 4f 53 20 56 34 20 32 20 56 34 20 32 20 54 5.2 50 72 6f 43 6f 6e 4f 53 20 56 34 20 32 20 56 34 20 32 20 56 34 20 32 20 54 5.2 50 74 5.2 5.2 5.2 5.2 5.2 5.2 5.2 5.2 5.2 5.2	 Frame 7: 210 ł Ethernet II, 9 Internet Proto Transmission 0 	473 30 0 56	170 00 0 10	TCD	64 40264 20	1747 [Dev yew] e
B Ethernet II, Src: PhoenixC_09:21:64 (00:a0:45:09:21:64), Dst: Vmware_e7:ca:1b (00:0c:29:e7:ca:1b) B Internet Protocol Version 4, Src: 172.20.3.10, Dst: 172.20.0.56 B Transmission Control Protocol, Src Port: 20547, Dst Port: 49261, Seq: 15, Ack: 21, Len: 156 B Data (156 bytes) 0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34 0000 02 92 00 56 34 0000 02 92 00 56 34 0000 02 92 00 56 34 0000 02 92 00 56 34 0000 02 92 00 56 34 0000 00 00 000 00 000 000 000 000 00	 Ethernet II, S Internet Proto Transmission (bytes on wire (1680	bits), 210 bytes	captured (1680) bits) on interf;	ace O
 B Internet Protocol Version 4, Src: 172.20.3.10, Dst: 172.20.0.56 B Transmission Control Protocol, Src Port: 20547, Dst Port: 49261, Seq: 15, Ack: 21, Len: 156 B Data (156 bytes) 0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34 0040 2e 32 50 72 6f 43 6f 6e 4f 53 20 56 34 2e 32 2e 	 Internet Proto Transmission (Src: PhoenixC_09:21:	:64 (00:a0:45:09:2	1:64), Dst: Vm	ware_e7:ca:1b (00	0:0c:29:e7:ca:1b)
B Transmission Control Protocol, Src Port: 20547, Dst Port: 49261, Seq: 15, Ack: 21, Len: 156 p Data (156 bytes) 0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34	Transmission @	cocol Version 4, Src:	: 172.20.3.10, Dst	: 172.20.0.56		
B Data (156 bytes) 0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34		Control Protocol, Sr	'c Port: 20547, Ds	t Port: 49261,	Seq: 15, Ack: 23	1, Len: 156
0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34	🖪 Data (156 byte	tes)				
0030 7f ec a9 9a 00 00 cc 01 c4 00 00 02 92 00 56 34						
0040 <u>2e 32 50 72 6f 43</u> 6f 6e 4f 53 20 56 34 2e 32 2e .2ProCon os v4.2.	0030 7f ec a9 9	9a 00 00 cc 01 c4 0	<u>00 00</u> 02 92 00 56	34	V4	
	0040 2e 32 50 7	72 6f 43 6f 6e 4f 5	3 20 56 34 2e 32	2e .2ProCon	05 V4.2.	

42

Phoenix Contact, the verdict

By sending A LOT of data and investigating the resulting packets, I got to deduce the necessary packets

Resulting in a script to control PLCs

Just replaying only the necessary packets, while changing some parameters

Reading out the PLC state + controlling it



initMonitor(s): send and recv(s,1000,'010000000002f0000000000000cfff41 64652e52656d6f74696e672e53657276696365732e49 send and recv(s,1000,'010000000 send and recv(s,1000,'0100) send and recv(s,1000,'010000000002a000000000000000d4ff4164652e52656d6f74696e672e53657276696365732e4944657 send and recv(s,1000,'010000000 send and recv(s, 1000, '01000000 send and recv(s,1000,'01000000000240000000000000000004164652e52656d6f74696e672e53657276696365732e49466f send and recv(s,1000,'010000000 send and recv(s,1000,'01000000 send and recv(s,1000,'0100000000002a00000000000000d4ff4164652e52656d6f74696e672e53657276696365732e4944657669636549 send and recv(s,1000,'010000000002900000000000000000004164652e52656d6f74696e672e5365727 send and recv(s,1000,'010000000 send and recv(s, 1000, '0100000000029000000000000000004164652e52656d6f74696e672e53657276696365732e4944617 send and recv(s,1000,'010000000002a send and recv(s,1000,'01000000 send and recv(s,1000, '0100000000028000000000000000d6ff4164652e52656d6f74696e672e53657276696365732e4943616c6c737461 send and recv(s,1000,'0100000000025000000000000000004164652e52656d6f74696e672e53657276696365732e49446562 send and recv(s,1000,'010000000 send and recv(s,1000,'010000000002e0000000000000000004164652e52656d6f74696e672e53657276696365732e495072 send and recv(s,1000,'0100000000)00000ceff4164652e52656d6f74696e672e53657276696365732e4953696d send and recv(s, 1000, '010000000 000000000004164652e52656d6f74696e672e53657276696365732e4953696d706c6546 send and recv(s,1000,'010002000000e00030003000000000000000012401340130011401200') return

Beckhoff, the verdict

Beckhoff uses a software implementation called "Twincat", the underlying protocol is called AMS and uses the concept of "routes" to allow PCs to program the PLC. A PLC without routes does not listen!



Siemens, the verdict

Siemens uses S7Comm, which has already been researched closely, e.g. in snap7, scan7 or plcscan

However, every ICS Vendor uses some kind of broadcast protocol to easily find their devices on any network.

-> For Beckhoff, it would be the aforementioned UDP protocol -> For Siemens, this is called Profinet Discovery Protocol, known and understood by Wireshark ③

So I wrote an all-in-one script ...

- -1- And we noticed that not only PLCs respond but also Siemens Routers, Switches, HMI's and also PhoenixContact PLC's, IO Islands etc ...
- -2- There is a Profinet DCP GET, but also Profinet DCP SET. Which allows, a.o., IP data to be set, ALSO on Switches and Routers !!

Pentesting ICS

Start to Pentest



Pentesting ICS

>Not as easy as it sounds

- ICS hardware (HMI, PLC, OPC ...) has very, very limited hardware
- These devices cannot handle an excessive amount of traffic

For example (<u>http://csrc.nist.gov/publications/nistpubs/800-82/SP800-82-final.pdf</u>):

- In a factory for integrated circuits, a ping sweep caused a 3M robotic arm to swing out of control
- Natural gas distribution utility, a pen tester went **a little bit** out of scope and caused a gas blockage to the customers for four (4!) hours

Can I still scan?

Yes, even Nmap can still be used and here is why:
 ➢ Reduce the scanning speed! Use "--scan-delay=1" to scan only one port at a time

Do not perform ping, SYN or UDP scans, but stick to TCP scans
 nmap -sn -PR -n

No fingerprinting options, these will send a lot of extra packets to force the listener to respond with certain crafted packets. These often contain "fingerprints" that nmap compares with a built-in database to determine services and operating systems.

There is always another way

It is always to use the tools that are designed to work on a network \bigcirc \rightarrow On other words: use your head

Example of such a protocol:

SNMP!

- Simple Network Management Protocol is a technology designed to gather information on "your" network. (SNMP GET)
- > It can sometimes even be used to configure certain devices (SNMP SET)
- Since devices that allow this traffic are supposed to be able to handle the load it is a more or less "safe" way of scanning a network
 - Sometimes with surprising results ^(C)

SNMP HowTo

- 3 versions of which SNMPv1 is too old and SNMPv3 is too new to be seen regularly.
 - SNMPv3 supports credentials, but is hardly used nor supported in older hardware
- > So SNMP version 2*c* is the main focus.
 - Good and Free Windows Tool to get and set data for a given device:
 - MIB Browser (<u>http://ireasoning.com/mibbrowser.shtml</u>)
 - But off course extensive scanners are out there as well:
 - snmpwalk, for <u>Windows</u> or Linux (apt-get install snmp)
 - And Home Made Scripts ^(C)



As seen earlier today: No Focus On Security

And ICS itself?

- Most common ICS protocols like ProfiNet, Modbus and S7Comm do not even support encryption and sometimes not even authentication.
- Easily confirmed: for any PLC or HMI, just download the official programming software (e.g. *trial*). Install it and start connecting to and controlling the PLC!

ICS security tools?

 \succ Yes, a lot of specialized tools: PLCScan: http://www.digitalbond.com/tools/plcscan/ by http://scadastrangelove.org/ Only scans ports 102 (Siemens S7Comm) & 502 (Modbus) Mbtget for ModbusTCP: https://github.com/sourceperl/mbtget Stop/Run PLC's can be done with modicon command Logic down- and upload can be done with modicon stux transfer Profinet, IEC, S7-1200 scripts: https://github.com/atimorin/scada-tools

➢Some are built into metasploit ...

➤And we have some of our own as well ☺

msf auxiliary(modbusdetect) > show info

Name: Modbus Version Scanner

<u>msf</u> auxiliary(modbusdetect) > use auxiliary/scanner/scada/ use auxiliary/scanner/scada/digi addp reboot use auxiliary/scanner/scada/digi addp version ^{pr}use auxiliary/scanner/scada/digi realport serialport scan use auxiliary/scanner/scada/digi realport version ^{Ba}use auxiliary/scanner/scada/indusoft_ntwebserver_fileaccess use auxiliary/scanner/scada/koyo login use auxiliary/scanner/scada/modbus findunitid use auxiliary/scanner/scada/modbusclient use auxiliary/scanner/scada/modbusdetect use auxiliary/scanner/scada/sielco winlog fileaccess Description: This module detects the Modbus service, tested on a SAIA PCD1.M2 system. Modbus is a clear text protocol used in common SCADA systems, developed originally as a serial-line (RS232) async protocol, and later transformed to IP, which is called ModbusTCP. References: http://www.saia-pcd.com/en/products/plc/pcd-overview/Pages/pcd1-m2.aspx

http://en.wikipedia.org/wiki/Modbus:TCP

All right, time to hack ...euh... pentest ...euh... audit!

Methodology

➤Scan

find all IP addresses and open ports on a network

➢Enumerate

find OS and Service versions, create connections

Vulnerability assessment

• use specific vulnerability scanners

➢ Exploit

gain unauthorized access to a system

➢Gather

reuse found data/credentials on earlier services (and repeat)

Find all IP addresses and ports



Nmap (Linux/Windows) is our preferred tool here

Zenmap is the GUI, but you lose control on what exactly is happening

E.g. nmap -n -sP 10.0.0/24

- Use the resulting MAC addresses (if possible) to determine the device types
- E.g. nmap -n -sT 10.0.0.4,6,10-20
 - The port number gives you an idea what service it is
- Important: use your head
 - what would be your reaction if you see an open port 80?
 - what would you try if you see an open port **21**?

Thinking out of the box (scanning)



Devices such as Routers, Switches, Printers, Access Points, Card Readers, Airconditioners,

- PDU's ... might also have an IP address!
- So don't limit your scan to a single network...
- Which networks?
 - Use tracert -d (Windows) or traceroute -n (Linux)
- > Make sure to also including scanning the **default** network range for most of the above devices:
 - 192.168.1.0/24 and 192.168.0.0/24
 - They may be forgotten about upon installation I → E.g. Managed Switch still has IP 192.168.1.1 and default credentials, but has a lot of **VLAN** information





Again, Nmap can make connections and perform basic version detection
 E.g. nmap -n -sV -p102,3389,443,445 10.0.0.4,6,10-20

- This may even already disclose vulnerabilities, e.g. if you find SMB version 5.1.2600 (WinXPSP2)
- Can even run scripts (C:\Program Files (x86)\Nmap\scripts) E.g. nmap --script=tftp-enum,vnc-info 10.0.0.1-20
 - There are more scripts online
 - Siemens Scada: https://github.com/drainware/nmap-scada
 - ICS: <u>https://github.com/digitalbond/Redpoint</u>

Thinking out of the box (enumeration)

- The most common services can be enumerated
 SNMP (see earlier)
- DNS (zone transfer, anyone?)
- NetBIOS \rightarrow
- Active Directory / LDAP
 - (E.g. with <u>ridenum</u>)
- SMTP (nmap --script=smtp*)

Supe	erScan 4.0						
	Scan Host and Service Disc	overy Scan Options	Tools	Windows	Enumeration Abo	ut	(
	Hostname/IP/URL vwin2	:000			<u>E</u> numerate	Options	<u>C</u> lear
SuperScan 4 by Foundstone	Enumeration Type NetBIOS Name Table NULL Session MAC Addresses Vorkstation type Users Groups RPC Endpoint Dump Account Policies Shares Domains Remote Time of Day Logon Sessions Drives Trusted Domains Services Registry	VW1N2000 INet~Services IS~VW1N2000 ADMINISTRATOF WORKGROUP 1MSBROWSE_ MAC address 0: Attempting a N NULL session s MAC address 0: \Device\Net 227CBD5C1258} Workstation/se Windows 2000 Workstation/Se Platform ID Version Comment	03 1C 00 03 1D 01 00:00 ULL se uccess on 172 00:00 biosSm 00:00 BT_Tcp erver t	UNIQUE GROUP UNIQUE UNIQUE GROUP ::29:E2:6 :ssion co ful to \ .23.83.2 :00:00:0 :29:E2:6 :ip_{7C0E ::29:E2:6 :ip_{7C0E ::29:E2:6 :ip_{7C0E ::29:E2:6 ::5. : ""	Messenger na Domain contr Workstation Messenger na Master brows A:47 nnection on 1 \172.23.83.23 3 0:00 A:47 5066-950D-450 72.23.83.23 72.23.83.23 0 0	me coller name service name me rame 72.23.83.23 ClPC\$	E
:02	Saved log file	Live: 0	TCP op	en: 0	UDP open: 0	1/1 done	

Vulnerability Assessment

Finding vulnerabilities can be automated
➤ Tenable Nessus Vulnerability Scanner
➤ Has a free version (16IP limitation)
➤ Some (ICS) companies run Nessus on a weekly base

- Delivers beautiful reports
- Can be configured with scan delays
- Detailed information on vulnerabilities and potential exploitability



Thinking out of the box (vulnerabilities)

- > There are also **protocol vulnerabilities**:
 - DNS
 - HTTP
 - TELNET
 - FTP
 - SMTP

...

- ... all have one thing in common: they're cleartext protocols!
 Which means: they can be MiTM'ed and sniffed easily
 - Windows-tool: Cain & Abel, Linux-tool: ettercap



- SQL Injection on websites?
- Buffer Overflow on Windows SMB?
- Profinet Set on PLC?
- There is a good tool for that: the Metasploit Framework

Exploiting

- Has a community edition (free), and exists for both Windows (buggy) and Linux (better)
- Has a lot of built in exploits (currently around 1540), scanners and modules
- There are also online exploit databases like <u>www.exploit-db.com</u>

Mmetasploit[®]

Thinking out of the box (exploitation)

Just logging into a device without permission is also considered exploitation

Not only devices car (and they often are)

- Dropping USB Key
- Phishing emails
- Shoulder surfing
- Or ---->



TV5MONDE : CYBERATTAQUE INÉDITE

"On est heureux de revenir à l'antenne" mais "on est loin de triompher" (directeur général de TV5Monde sur BFMTV).

1.40%

Gather		
Guarci	<pre>root@kalitijl:~/Desktop# l</pre>	s /usr/share/metasploit-framework/modules/post/windows
	arp scanner.rb	enum ms product keys.rb
	bitcoin_jacker.rb	enum_muicache.rb
The main goal for ma	ct	enum_patches.rb
root@kalitijl:~/Desktop#	ls /usr/share/metasploit	-framework/modules/post/windows
/gather/credentials/		
bulletproof_ftp.rb	gpp.rb	skype.rb
coreftp.rb	idm.rb	smartermail.rb
credential_collector.rb	imail.rb	smartftp.rb
domain_hashdump.rb	imvu.rb	spark_im.rb
dyndns.rb	mcafee_vse_hashdump.rb	sso.rb
enum_cred_store.rb	meebo.rb	steam.rb
enum_laps.rb	mremote.rb	tortoisesvn.rb
enum_picasa_pwds.rb	mssql_local_hashdump.rb	total_commander.rb
epo_sql.rb	nimbuzz.rb	trillian.rb
filezilla_server.rb	outlook.rb	vnc.rb
flashfxp.rb	razer_synapse.rb	windows_autologin.rb
ftpnavigator.rb	razorsql.rb	winscp.rb
ftpx.rb	<u>r</u> dc_manager_creds.rb	wsftp_client.rb
	enum_hostfile.rb	win_privs.rb
	enum_logged_on_users.rb	word_unc_injector.rb

A 100 A 400

STREET OF COL

- Just looking through files, browser history and stored passwords is one thing the aforementioned modules can assist with
- However nothing beats a manual search through some text files (*password.txt* on the desktop?), installing keyloggers or going for software like KeePass.
- And then there is also the PC's Memory
 - Data stored inside PC memory is generally not encrypted
 - It is a default setting for Windows to store all Kerberos and Windows local passwords in memory for future authentication (and backwards compatibility)
 - <u>Mimikatz</u> is a software that can extract these secrets
- Don't underestimate the importance of password hashes

The Workshop

Perform your own scanning on our *special* demonstration.

- Nothing is off limits, we have everything backed up
- Learn to use Nmap, in Windows or Linux, maybe extended with extra (ICS) scripts
- Feel free to try enumeration (recommended), vulnerability scanning and maybe even exploitation

- Inside this network we have also set up several Siemens S7-1200 PLC's connected to a functional setup, on every table.
 - There are several exercises, all explained on the local posters

Introducing our Factory



Fictile: the network



What can be done?

Office / datacenter (172.20.0.0 /16)



➢ Everything off course ☺

- The AD Environment is completely vulnerable
 - Don't go directly for the Domain Controller, but it is the ultimate goal
 - Find the weak spot!

> Can you print any text on the network printer? (One command!)

> There is also a vulnerability on the WiFi AP (on the LAN side)

And ICS Hacking?



For PhoenixContact; there are misconfigurations and vulnerabilities
 No special scripts needed for some of these
 Beckhoff has several vulnerabilities in both the HMI as the PLC
 Special scripts may be needed but are available
 Siemens has the most common configuration