TippingPoint a division of 3Com

SCADA Protocol Fuzzer & The Next generation of Inline Devices

DVLabs

- Ganesh Devarajan

TippingPoint a division of 3Com

Agenda

- Introduction to SCADA networks
  - Overview
  - SCADA Protocols
- SCADA Security
  - Attack scenarios
  - Past known attacks
- SCADA Fuzzer
- Next Generation of Inline Devices
- Demo

TippingPoint a division of 3Com **SCADA Definition**

SCADA

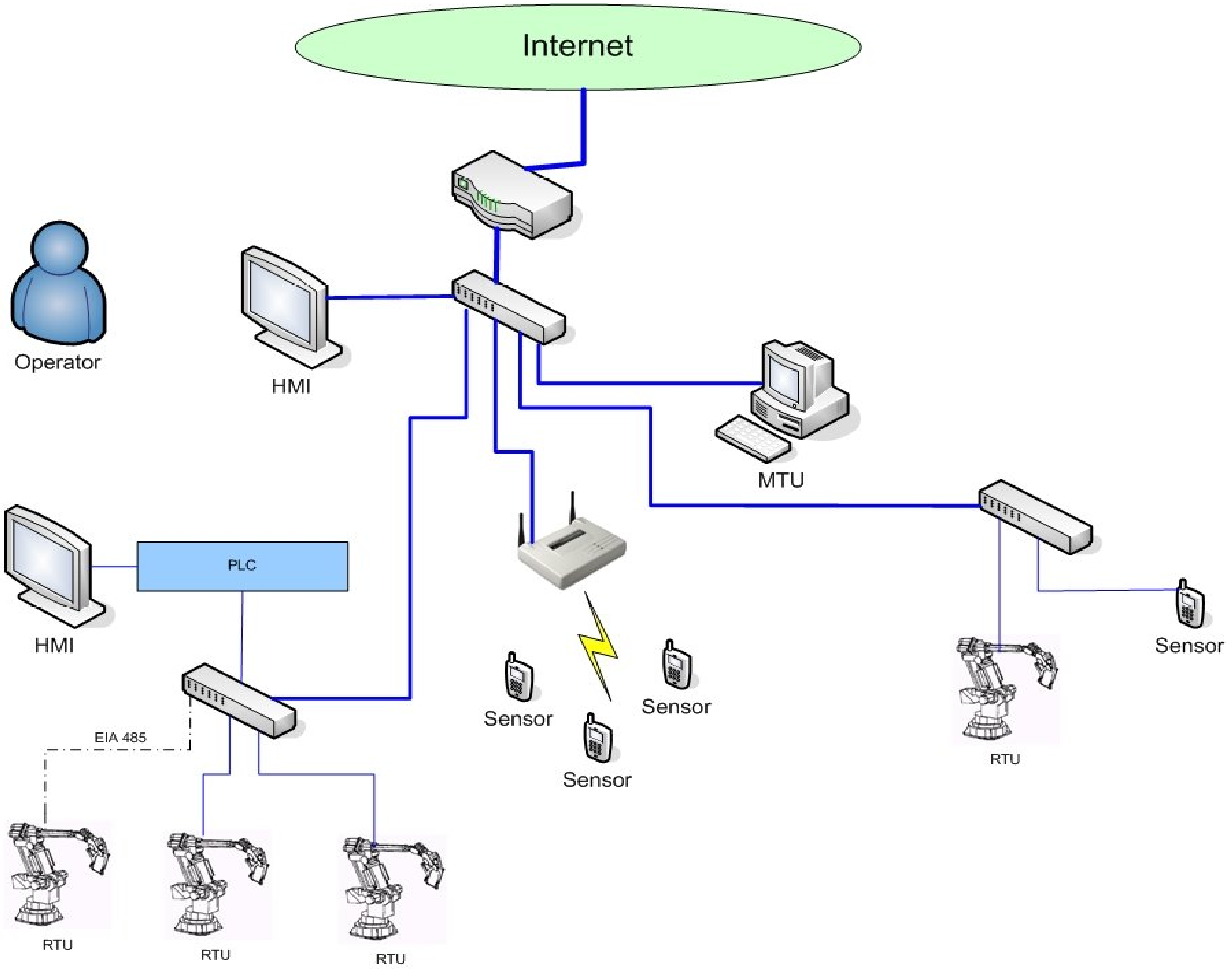
- ***Supervisory Control and Data Acquisition*** is defined as a common process control application that collects data from sensors on the shop floor or in remote locations and sends them to a central computer for management and control.
- It is the vital component of any Critical Infrastructure.
- They are used for sensing/managing real-time data
  - Water
  - Gas
  - Electricity
  - Refineries
  - Nuclear plants
  - Other manufacturing operations.

SCADA Infrastructure

- SCADA System Components
  - Operator
  - Human Machine Interface (HMI)
    - Presents data to the user
    - GUIs, Schematics, Windows
  - Master Terminal Unit (MTU)
    - Processes the data and presents it to HMI
  - Communication channel
    - Internet, wireless, switched network, etc
  - Remote Terminal Unit (RTU)
    - Abstracts data and sends it to MTU

TippingPoint a division of 3Com

SCADA Infrastructure



TippingPoint a division of 3Com

Need for SCADA Security

The need for security in SCADA systems

- When these protocols were initially created they were proprietary and were not linked to the outside world. But with the improved communication protocols they were exposed more to the Internet. The systems that control our day to day living is exposed to the outside world without any inbuilt security features.
- It is easier to take down the entire country's Critical Infrastructure.
  - Black out
- On a smaller scale you can take down the company's manufacturing plant.
  - The cooling system of the Server room
  - False reports at the manufacturing plant

SCADA Attack Scenarios

- Providing False Data - The functionality of the RTU is to either read or write data into the server and the compromised RTU can write false data into the server.
  - Sensors for Water pollutants
  - Temperature sensors in server rooms
- Denial of Service Attack
  - Continuous sting of reboot command
- Protocol anomalies

SCADA Attacks

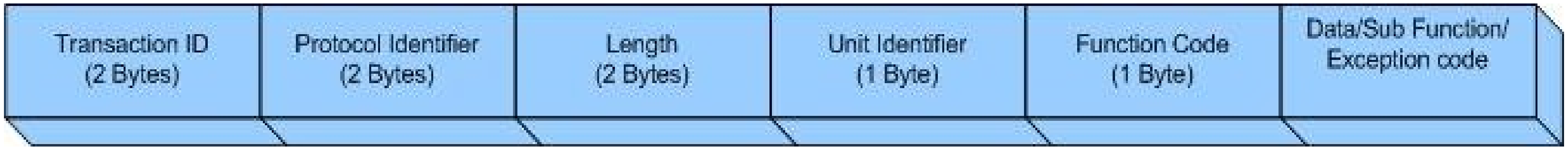
- Cyber-Attacks by Al Qaeda Feared
    - Washington Post, June 27, 2002 Mountain View, Calif
- Information-technology contractor Vitek Boden who used his knowledge of control systems to release millions of liters of sewage into drinking water
- Slammer worm affected the operation of the corporate network at Ohio's inactive Davis-Besse nuclear plant and disabled a safety monitoring system for nearly five hours in January 2003
- An hacker took control of the gas pipelines run by Gazprom for around 24 hours in 1999 in Russia

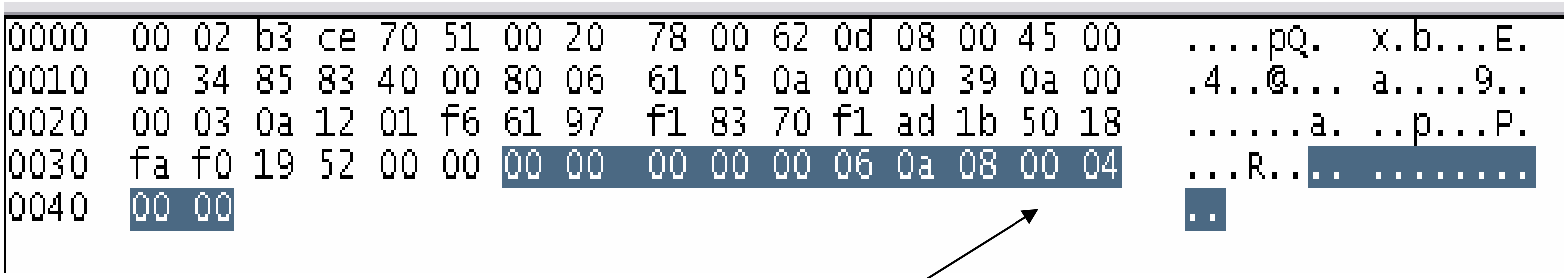
SCADA Protocols

- Modbus
- DNP3
- ICCP
- UCA 2.0 and IEC 61850 Standards
- Control Area Networks
- Control Information Protocol
- DeviceNet
- ControlNet
- OPC
- Profibus

SCADA Protocols

- MODBUS





```
0000  00 02 b3 ce 70 51 00 20  78 00 62 0d 08 00 45 00   ....pQ.  x.b...E.
0010  00 34 85 83 40 00 80 06  61 05 0a 00 00 39 0a 00   .4..@... a....9..
0020  00 03 0a 12 01 f6 61 97  f1 83 70 f1 ad 1b 50 18   ......a. ..p...P.
0030  fa f0 19 52 00 00 00 00  00 00 00 06 0a 08 00 04   ...R.... ........
0040  00 00                                              ..
```

Force Listen Mode

SCADA Protocols

Function Code	Function Name
01	Read Coil Status
02	Read Input Status
03	Read Holding Registers
04	Read Input Registers
05	Force Single Coil
06	Preset Single Register
07	Read Exception Status
09	Program 484
0A	Poll 484
0B	Fetch Communication Event Counter
0C	Fetch Communication Event Log
0D	Program Controller
0E	Poll Controller
0F	Force Multiple Coils
10	Preset Multiple Registers
11	Report Slave ID
12	Program 884/M84
13	Reset Communication Link
14	Read General Reference
15	Write General Reference
16	Mask Write 4X Register
17	Read/Write 4X Registers
18	Read FIFO Queue

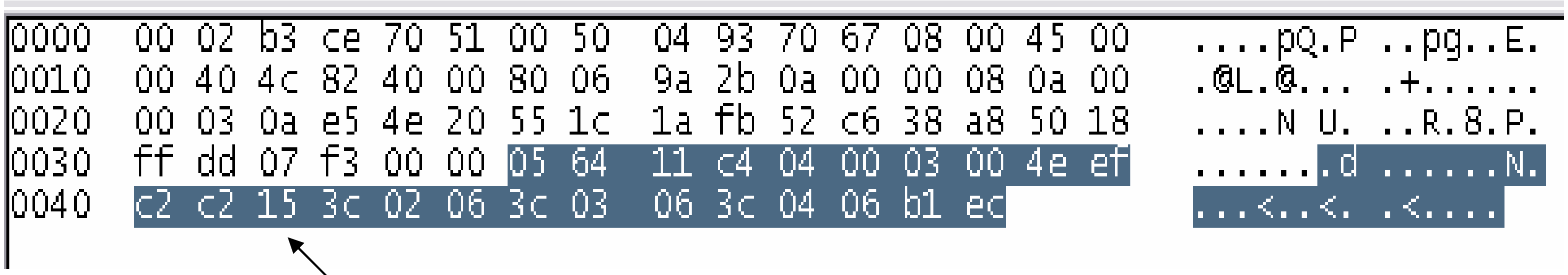
SCADA Protocols

Function Code	Sub-Function Code	Function Name
08	00	Return Query Data
08	01	Restart Communication Option
08	02	Return Diagnostic Register
08	03	Change ASCII Input Delimiter
08	04	Force Listen Only Mode
08	05-09	Reserved
08	0A	Clear Counters and Diagnostic Reg.
08	0B	Return Bus Message Count
08	0C	Return Bus Communication Error Count
08	0D	Return Bus Exception Error Count
08	0E	Return Slave Message Count
08	0F	Return Slave No Response Count
08	10	Return Slave NAK Count
08	11	Return Slave Busy Count
08	12	Return Bus Char. Overrun Count
08	13	Return Overrun Error Count
08	14	Clear Overrun Counter and Flag
08	15	Get/Clear Modbus Plus Statistics
08	16-UP	Reserved

SCADA Protocols

- DNP3

Sync (2 Bytes)	Length (1 Byte)	Link Control (1 Byte)	Destination Address (2 Bytes)	Source Address (2 Bytes)	CRC (2 Bytes)	Data



```
0000   00 02 b3 ce 70 51 00 50  04 93 70 67 08 00 45 00   ....pQ.P ..pg..E.
0010   00 40 4c 82 40 00 80 06  9a 2b 0a 00 00 08 0a 00   .@L.@... .+......
0020   00 03 0a e5 4e 20 55 1c  1a fb 52 c6 38 a8 50 18   ....N U. ..R.8.P.
0030   ff dd 07 f3 00 00 05 64  11 c4 04 00 03 00 4e ef   .......d ......N.
0040   c2 c2 15 3c 02 06 3c 03  06 3c 04 06 b1 ec        ...<..<. .<....
```

Disable Spontaneous messages

SCADA Protocols

- Control Byte
  - Control function code
- Transport Layer byte
  - First-Final
  - Sequence Number
- Application Layer Control Byte
  - First-Final
  - Confirm
  - Sequence
- Data chunking
  - CRC DNP
  - 2 CRC bytes Every 16 bytes of data

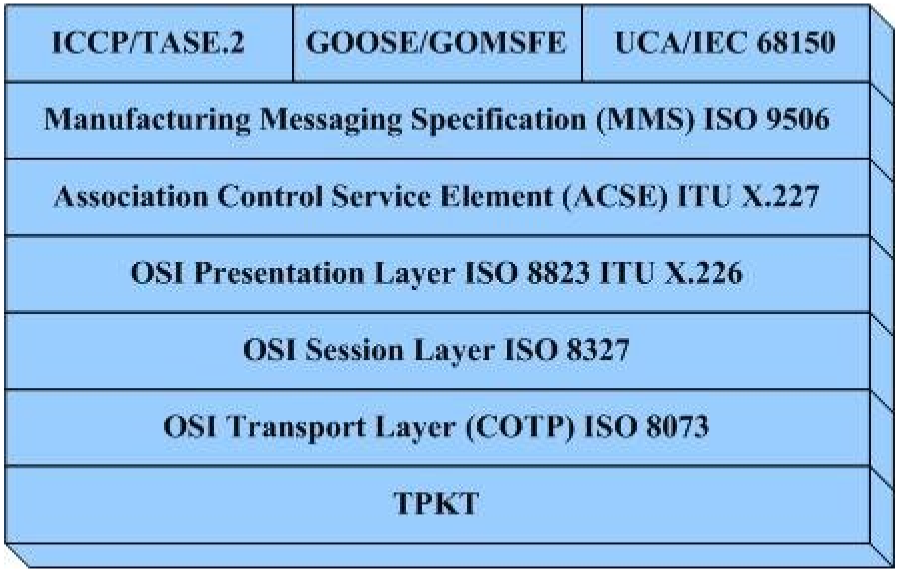
TippingPoint
a division of 3Com

SCADA Protocols

Bit	Internal Indication Flag
0	Last received message was Broadcast message
1	Class 1 Data available
2	Class 2 Data available
3	Class 3 Data available
4	Time Synchronization Required
5	Digital Output in Local
6	Device Trouble
7	Device Restarted
8	Function Code (Not Implemented)
9	Requested Object Unknown or Application Error
10	Parameters Out of range
11	Even buffer overflowed
12	Operation already executing
13	Configuration Corrupt
14	Not used (returns 0)
15	Not used (returns 0)

SCADA Protocols

- ICCP



ICCP/TASE.2	GOOSE/GOMSFE	UCA/IEC 68150
Manufacturing Messaging Specification (MMS) ISO 9506		
Association Control Service Element (ACSE) ITU X.227		
OSI Presentation Layer ISO 8823 ITU X.226		
OSI Session Layer ISO 8327		
OSI Transport Layer (COTP) ISO 8073		
TPKT		

TippingPoint SCADA Fuzzer

- What does the SCADA Fuzzer detect?
  - Protocol anomalies
  - Unauthorized client/server communication
  - Unauthorized client/server command execution
  - Possible Denial of Service attacks
- What protocols are we covering today?
  - MODBUS
  - DNP3

TippingPoint a division of 3Com

SCADA Fuzzer

- Fuzzer Components
  - __init.py – Defines all the aliases
  - blocks.py – Defines blocks and block helpers
  - pedrpc.py – Communication purposes and an interface with the main fuzzer
  - primitives.py – the fuzzer primitives includes string, static, etc
  - sessions.py – Functionality for building and executing session
  - sex.py – Sulley's exception Handler
- Agents
  - network_monitor.py – Monitors network communications and logs the pcap files
  - process_monitor.py – Detects the faults
  - vmcontrol.py – Interfaced with the VM image to start, stop, suspend and reset the image along with deleting and restoring the snapshots

MODBUS Code Snipet

```
s_initialize("MODBUSFUNCCODE01")
  # Transaction ID
  s_static("\x00\x01")
  # Modbus Protocol Identifier
  s_static("\x00\x00")
  # Length bytes
  s_sizer("modlength", length=2, name="length", endian=">", fuzzable=False)
  if s_block_start("modlength"):
     # Unity Identifier
     s_static("\x0D")
     # Function Code
     s_byte(0x01)
     # Data or Sub function Code
     s_dword(0x00000000)
  s_block_end()
```

TippingPoint a division of 3Com

DNP3 Code Snipet

Static Length

```
s_initialize("DNP3StaticLength")
if s_block_start("header"):
  s_static("\x05\x64") # Start Sync Bytes.
  # Length Bytes we are having it as a constant length at first
  s_static("\x12")
  # Control Byte
  s_byte(0xc4, full_range=True)
  # Destination Address
  s_short(0x0400)
  # Source Address
  s_short(0x300)
  s_block_end()
# Checksum of the DNP Header.
s_checksum("header", algorithm=dnp_crc16, length=2)

# The Data POrtion of the Packet
if s_block_start("Data"):
  # Transport Layer Chunk
  s_byte(0xc2, full_range=True)
  # Application Chunk
  s_byte(0xc2, full_range=True)
  # Function Code
  s_byte(0x0d, full_range=True)
  # Static Data for now..
  s_static("AAAAAAA")
  # This will fuzz a huge array of string cases..
  s_block_end()
s_checksum("Data", algorithm=dnp_crc16, length=2)
```

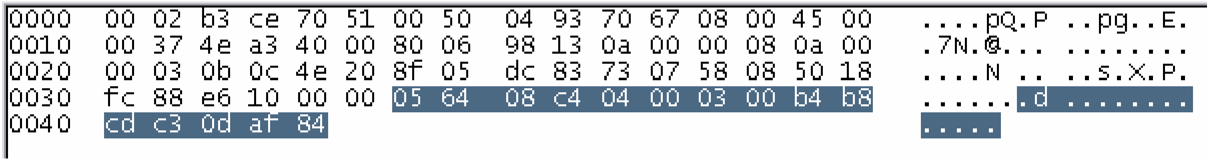
s_string(“A”) + Chunkdnp3(data)

TippingPoint
a division of 3Com

The Next Generation of Inline Devices

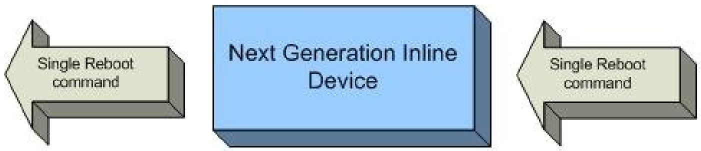
The Next Generation of Inline Devices

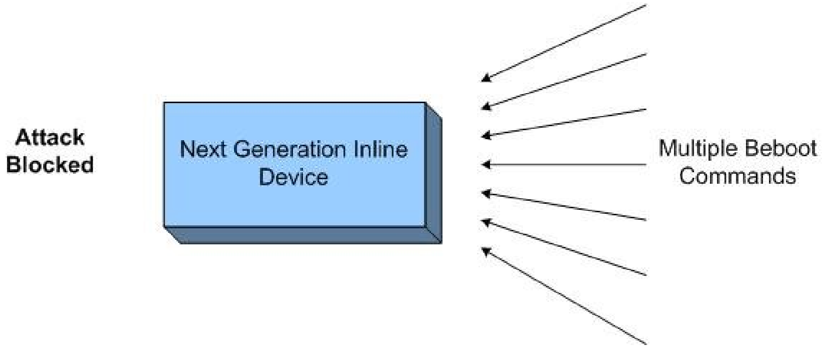
- Reboot command

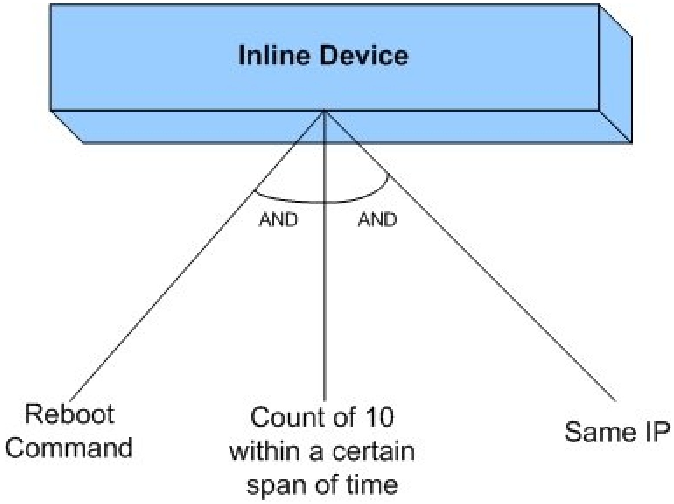


```
0000   00 02 b3 ce 70 51 00 50  04 93 70 67 08 00 45 00    ....pQ.P ..pg..E.
0010   00 37 4e a3 40 00 80 06  98 13 0a 00 00 08 0a 00    .7N.@... ........
0020   00 03 0b 0c 4e 20 8f 05  dc 83 73 07 58 08 50 18    ....N .. ..s.X.P.
0030   fc 88 e6 10 00 00 05 64  08 c4 04 00 03 00 b4 b8    .......d ........
0040   cd c3 0d af 84                                      .....
```

- \x0d is the Cold Reboot command in the DNP3 protocol
  - Just one of those could be legitimate







TippingPoint
a division of 3Com

Demo

TippingPoint a division of 3Com

References

- The SCADA Architecture and basic implementation details: ***Securing SCADA Systems – Ronald L. Krutz. PhD***
- Modbus***: www.modbus.org***
- DNP3***: www.dnp3.org***
- ICCP***: www.iccp.org***
- Attack Details**:** ***www.digitalbond.com***
- Modbus Protocol details***: http://www.modbustools.com/PI_MBUS_300.pdf***
- DNP3 Protocol Primer***: http://www.dnp.org/About/DNP3%20Primer%20Rev%20A.pdf***
- DNP3 User and Reference Manual by Control Microsystems***: https://dg.controlmicrosystems.com/Technical%20Support/Software,%20Manuals%20and%20Release%20Notes/Protocols/DNP3%20Protocol/Manuals/DNP3_User_and_Reference_Manual.pdf***
- ICCP Guide***: www.sisconet.com/downloads/usrguid5.doc***
- Matt Franz Wiki***: http://www.scadasec.net/secwiki/SecProducts***

TippingPoint
a division of 3Com

Acknowledgements

- Pedram Amini and Cody Pierce for developing the Sulley Fuzzing Framework

TippingPoint
a division of 3Com

Thank you

Ganesh Devarajan

-Ganesh_Devarajan@3com.com