Network Security by Cisco

Slavonice, 28th June 2007

Petr Matoušek

`matousp@fit.vutbr.cz`

Roadmap

Motivation

1. **Basic Principles of Intrusion Detection Systems**
   - **Signature-Based Detection**
   - **Anomaly-Based Detection**
2. **Network Security by Cisco**
   - **Introduction Cisco Context-Based Access Control (CBAC)**
   - **Flexible Packet Matching (FPM)**
   - **Cisco Secure Monitoring, Analysis and Response System (MARS)**
   - **Cisco Self-Defending Networks (Cisco SDN)**
3. **Advanced Techniques for Traffic Analysis**
   - **Cisco Service Control Engine (SCE)**
   - **ModSecurity – HTTP security**

Motivation

Motivation

❖ Botnet – a new phenomenon in attacking strategy

- **botnet –** *“bot-net”, “robot-network”, “software robots”*
- **a collection of compromised computers under common control**
- **used for sending spams, DDoS attacks, Phishing, Theft of Identity, etc.**

Motivation

❖ Botnet – a new phenomenon in attacking strategy

- **botnet –** *“bot-net”, “robot-network”, “software robots”*
- **a collection of compromised computers under common control**
- **used for sending spams, DDoS attacks, Phishing, Theft of Identity, etc.**

❖ Bots on rise

- **average of 10,352 active botnets per day (Symantec, 20056)**
- **DoS attacks: from 119 to 927 per day (last 6 month, Symantec, 2005)**
- **2005, Dutch police discovered a botnet of 1,5 milion zombie PCs**
- **DDos-for-Rent: 80-90 for average site, higher for more complicated**
- **Extortion: “You pay me 20,000 $ or your web site goes down!”**

Need for Network Security

Need for Network Security

- 1. Defend my own network

Need for Network Security

❖ **1. Defend my own network**

- **detect and isolate compromised host**
- **detect and stop sniffing, scanning (reconaissance), and access attacks**
- **detect and stop DDoS attack from inside/outside of the LAN/WAN**
- **complex solution over entire network**

Need for Network Security

❖ 1. Defend my own network

- detect and isolate compromised host
- detect and stop sniffing, scanning (reconaissance), and access attacks
- detect and stop DDoS attack from inside/outside of the LAN/WAN
- complex solution over entire network includes:
  - firewalls, IDS/IPS
  - antispam, anti-virus machines
  - monitors, collectors, managements stations
  - routers, switches, hosts

Need for Network Security

- **2. Secure my own services**

Need for Network Security

❖ 2. Secure my own services

- **classify/analyse network traffic (tunnels, dynamical ports)**
- **filter out bad traffic, pass legitimite one**
- **create/dynamically add my own rules and policies**
- **check application data to prevent attacks on application level**

A Road To Go

- **Challenges for research**

A Road To Go

❖ Challenges for research

- traffic analysis on **multigigabits networks** (e.g., signature detection)
- high-level protocol analysis – **application protocols**
- detection using **anomaly-based** behaviour
- rules describing protocols/attacks dynamically loaded to **FPGA**
- sophisticated analyses of different incidents, **corellation function**

A Road To Go

❖ Challenges for research

- traffic analysis on multigigabits networks (e.g., signature detection)
- high-level protocol analysis – application protocols
- detection using anomaly-based behaviour
- rules describing protocols/attacks dynamically loaded to FPGA
- sophisticated analyses of different incidents, corellation function

❖ Current security issues

- DDoS attacks
- WWW traffic, Emails
- IP telephony
- etc.

Research Background

Research Background

❖ Current Activites

- **Liberouter project – hardware acceleration on FPGA (CESNET)**
  - **FlowMon – passive network monitoring using FPGA**
  - **IDS – accelerated Network Intrusion Detection System**
  - **NetCOPE – rapid development of network applications**

Research Background

❖ Current Activites

- **Liberouter project – hardware acceleration on FPGA (CESNET)**
  - **FlowMon – passive network monitoring using FPGA**
  - **IDS – accelerated Network Intrusion Detection System**
  - **NetCOPE – rapid development of network applications**
- **Security-Oriented Research in Information Technology (FIT)**
- **Cisco Network Academy – Network Security on Cisco devices (FIT)**
  - **CCNA programme**
  - **Network Security programme (NS)**
  - **Fundamentals of Wireless Networks (FWL)**
- **BUSLab at FIT – Brno University Security Laboratory (FIT, FI)**

2 Basic Principles of Intrusion Detection Systems

2 Basic Principles of Intrusion Detection Systems

❖ 2.1 Signature-based detection

- IDS signatures identify and classify an alarm condition
- info or attack signatures
- incapable to detect new types of attacks

2 Basic Principles of Intrusion Detection Systems

❖ 2.1 Signature-based detection

- **IDS signatures identify and classify an alarm condition**
- **info or attack signatures**
- **incapable to detect new types of attacks**

❖ IDS signature classification

1. **based on number of packets needed for detection**
   - **atomic signatures – simple patters within a single packet**
   - **compound signatures – complex patterns within multiple packets**

2 Basic Principles of Intrusion Detection Systems

❖ 2.1 Signature-based detection

- IDS signatures identify and classify an alarm condition
- info or attack signatures
- incapable to detect new types of attacks

❖ IDS signature classification

1. based on number of packets needed for detection
    - atomic signatures – simple patters within a single packet
    - compound signatures – complex patterns within multiple packets
2. based on severity
    - information signatures – detect information-gathering activity
    - attack signatures – detect attacks into the protected network

2.1 Signature-based detection

❖ **Example – Snort rules:**

- **simple rule**

```
alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 111
(content:  "|00 01 86 95|")
```

2.1 Signature-based detection

❖ Example – Snort rules:

- **simple rule**

```
alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 111
(content:  "|00 01 86 95|")
```

- **ddos.rules -> set of 30 rules, example:**

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"DDOS TFN Probe";
icmp_id:678; itype:8; content:"1234"; reference:arachnids,443;
reference:cve, 2000-0138; classtype:attempted-recon; sid:221;
rev:5;)
```

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"DDOS tfn2k icmp
possible communication"; icmp_id:0; itype:0; content:"AAAAAAAAAA";
reference:arachnids,425; reference:cve,2000-0138;
classtype:attempted-dos; sid:222; rev:3;)
```

2.2 Anomaly-based detection

2.2 Anomaly-based detection

- Requires profiles for each user group
  - the profile defines the behaviour characteristics for a user group
  - the quality of the profiles directly relates to how successful IDS will be
- When a user changes behaviour, the IDS generate alarm

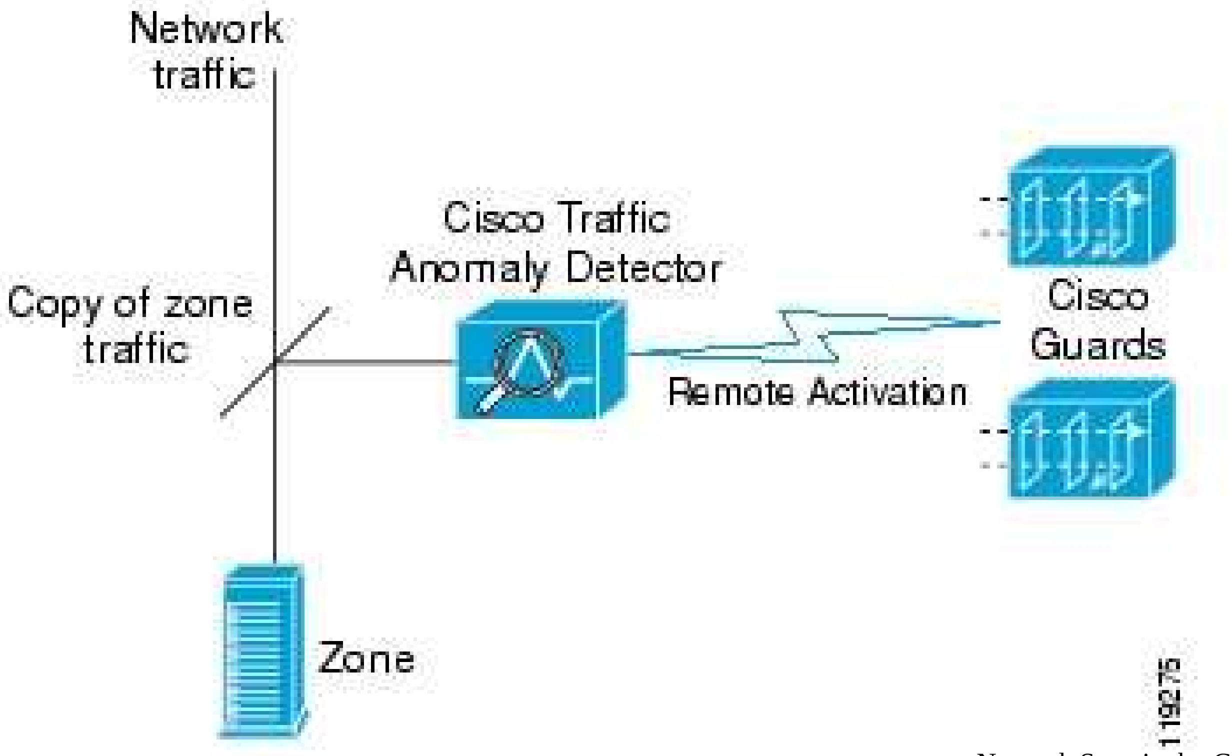
2.2 Anomaly-based detection

- **Requires profiles for each user group**
  - the profile defines the behaviour characteristics for a user group
  - the quality of the profiles directly relates to how successful IDS will be
- **When a user changes behaviour, the IDS generate alarm**
- **Advantages**
  - enables tunable control over false positives
  - detects previously unpublished attacks
- **Disadvantages**
  - require an initial training time
  - require updating user profiles as habits change
  - have difficulty correlating alarms to specific attacks

2.2 Anomaly-based detection

❖ Example – Cisco Traffic Anomaly Detector Module

- a hardware module that monitors a copy of the network traffic
- learns the zone traffic
- creates a set of zone-specific policies
- applies policies and detect anomalies
- effective for DDoS detection



2.2 Anomaly-based detection

- **The Learning Process**

2.2 Anomaly-based detection

❖ **The Learning Process**

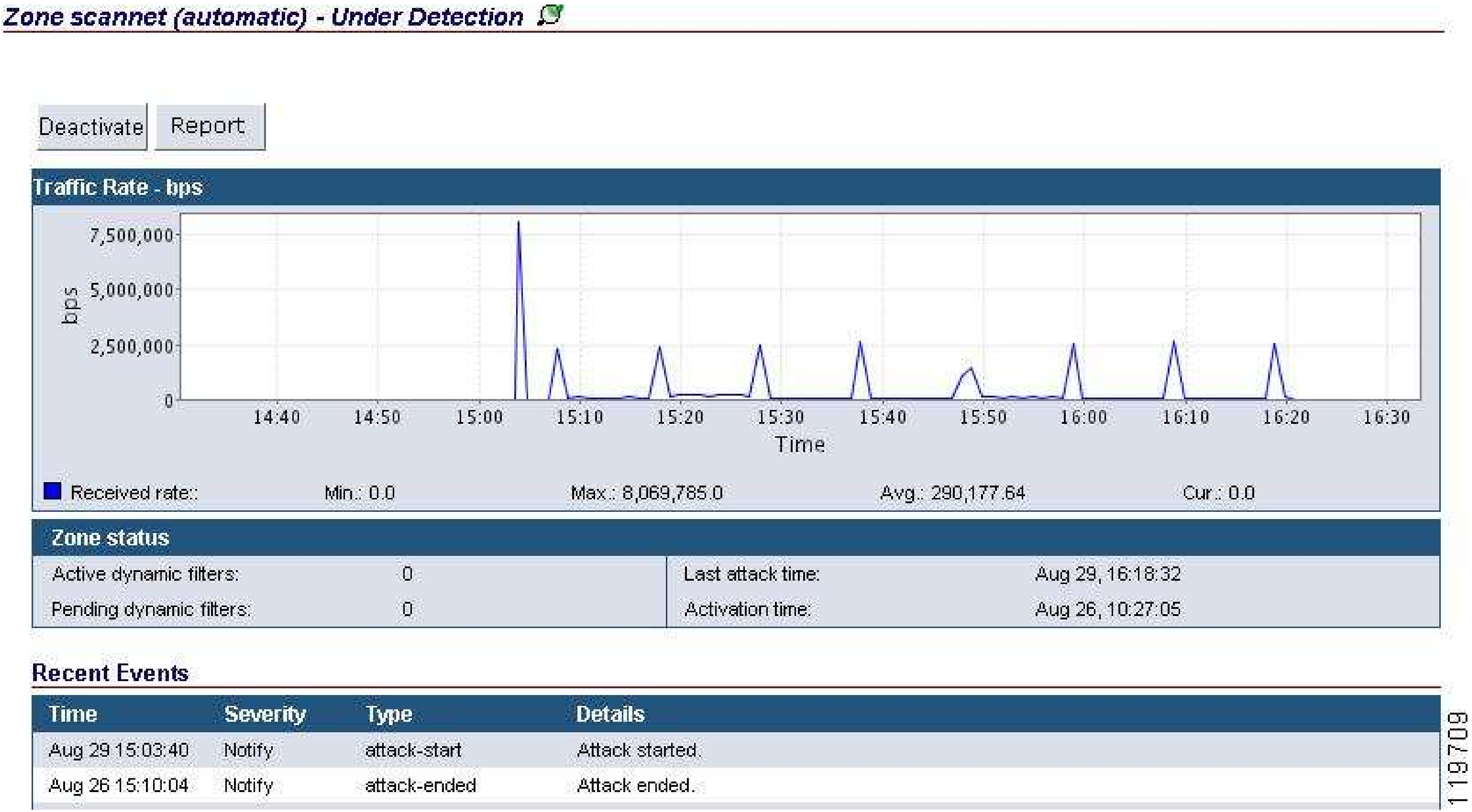
1. **policy construction**
   - **learns the characteristics (services and traffic rates) of the traffic**
   - **for both normal and peak traffic**
   - **detector creates policies based on the services**
   - **scans traffic flow ⇒ policy templates**
   - **modifies the default zone traffic policies and thresholds**

2.2 Anomaly-based detection

❖ The Learning Process

1. **policy construction**
   - **learns the characteristics (services and traffic rates) of the traffic**
   - **for both normal and peak traffic**
   - **detector creates policies based on the services**
   - **scans traffic flow ⇒ policy templates**
   - **modifies the default zone traffic policies and thresholds**
2. **threshold tuning phase**
   - **policy treshold exceeded → the detector executes action**

2.2 Anomaly-based detection



Zone scannet (automatic) - Under Detection

■ Received rate:: Min.: 0.0 Max.: 8,069,785.0 Avg.: 290,177.64 Cur.: 0.0

Zone status

Active dynamic filters:	0	Last attack time:	Aug 29, 16:18:32
Pending dynamic filters:	0	Activation time:	Aug 26, 10:27:05

Recent Events

Time	Severity	Type	Details
Aug 29 15:03:40	Notify	attack-start	Attack started.
Aug 26 15:10:04	Notify	attack-ended	Attack ended.

2.2 Anomaly-based detection

❖ Anomaly Detection Process: Traffic Filters

- **Bypass filters**
  - **prevent the Detector from applying rules**
  - **for specific traffic flows**
- **Flex-Content filters**
  - **traffic flow filtered according to the IP, TCP headers and the content**
- **Dynamic filters**
  - **apply the analysis detection level**
  - **anomaly detected → dynamic filters loaded**
  - **zone protection activated**

2 Basic Principles of Intrusion Detection Systems

❖ Conclusion – Current Issues of IDS

❖ Conclusion – Current Issues of IDS

- system limits: CPU performance, memory capacity, input data rates
- a huge number of alarms (false positives) generated by IDSs
- only narrow view on the network
- stateful behaviour – flows information needed
- application protocol analysis required
- mostly deployed signature-based detection only ⇒ a large set of rules
- tunnelling different protocols (e.g., over port 80)
- encrypted connections
- dynamic ports (multimedia)

3 Network Security by Cisco

3 Network Security by Cisco

- 3.1 Cisco Context-based Access Control (CBAC)
- 3.2 Flexible Packet Matching (FPM)
- 3.3 Cisco Security Monitoring, Analysis and Response System (MARS)
- 3.4 Cisco Self-Defending Networks (SDN)

3.1 Cisco Context-Based Access Control (CBAC)

3.1 Cisco Context-Based Access Control (CBAC)

❖ Features

- a type of ACLs (Access Control Lists)
- inspect traffic at layer 3 and higher
- manage state information for TCP and UDP sessions
- create temporary openings in the firewall

3.1 Cisco Context-Based Access Control (CBAC)

❖ How CBAC works

1. control traffic is inspected by the CBAC rule
2. creates a dynamic ACL to allow returning traffic throught the firewall
3. inspects control traffic, dynamically creates/removes ACLs
4. after session terminates CBAC removes all dynamic ACLs

3.1 Cisco Context-Based Access Control (CBAC)

❖ How CBAC works

1. control traffic is inspected by the CBAC rule
2. creates a dynamic ACL to allow returning traffic throught the firewall
3. inspects control traffic, dynamically creates/removes ACLs
4. after session terminates CBAC removes all dynamic ACLs

❖ TCP sessions

- CBAC checks TCP sequence numbers
- discards suspicious packets out of sequence
- monitors command channels only (FTP, SIP etc.)

3.1 Cisco Context-Based Access Control (CBAC)

❖ **DoS attack protection**

- number of half-open TCP connection
  - total number (default 500)
  - per time (one-minute high/low)
  - per host (default 50)

3.1 Cisco Context-Based Access Control (CBAC)

❖ DoS attack protection

- number of half-open TCP connection
  - total number (default 500)
  - per time (one-minute high/low)
  - per host (default 50)
- wait and idle times
  - SYN (30 sec to reach the established state)
  - FIN (session closed 5 sec after FIN)
  - idle times: TCP (1 hour), UDP (30 sec), DNS (5 sec)
- reactions
  - reset (RST) the oldest half-open connection
  - temporary block all incoming SYN packets

3.1 Cisco Context-Based Access Control (CBAC)

❖ CBAC-Supported Protocols

- **TCP, UDP, ICMP**
- **RPC, Unix R-commands**
- **FTP, TFTP, SMTP**
- **Java, SQL*Net, URL filtering**
- **RTSP, H.323**

3.1 Cisco Context-Based Access Control (CBAC)

❖ Conclusion

- a technique for data analysis on higher layers
- more sophisticated ACLs
- a part of router’s operating system IOS
- keep state information
- predefined rules and actions → easy to deploy
- supports limited fixed number of application protocols
- new attacks and protocols cannot be added

3.2 Flexible Packet Matching (FPM)

3.2 Flexible Packet Matching (FPM)

❖ Introduction

- define traffic classes and actions (policies) to block network attacks
- ACL pattern matching tool for thorough and customized packet filters
- provides match on arbitrary bits of a packet at arbitrary depth
- matches packet header + first 256 bytes of payload
- FPM provides a flexible layer 2-7 stateless classification mechanism

3.2 Flexible Packet Matching (FPM)

❖ Introduction

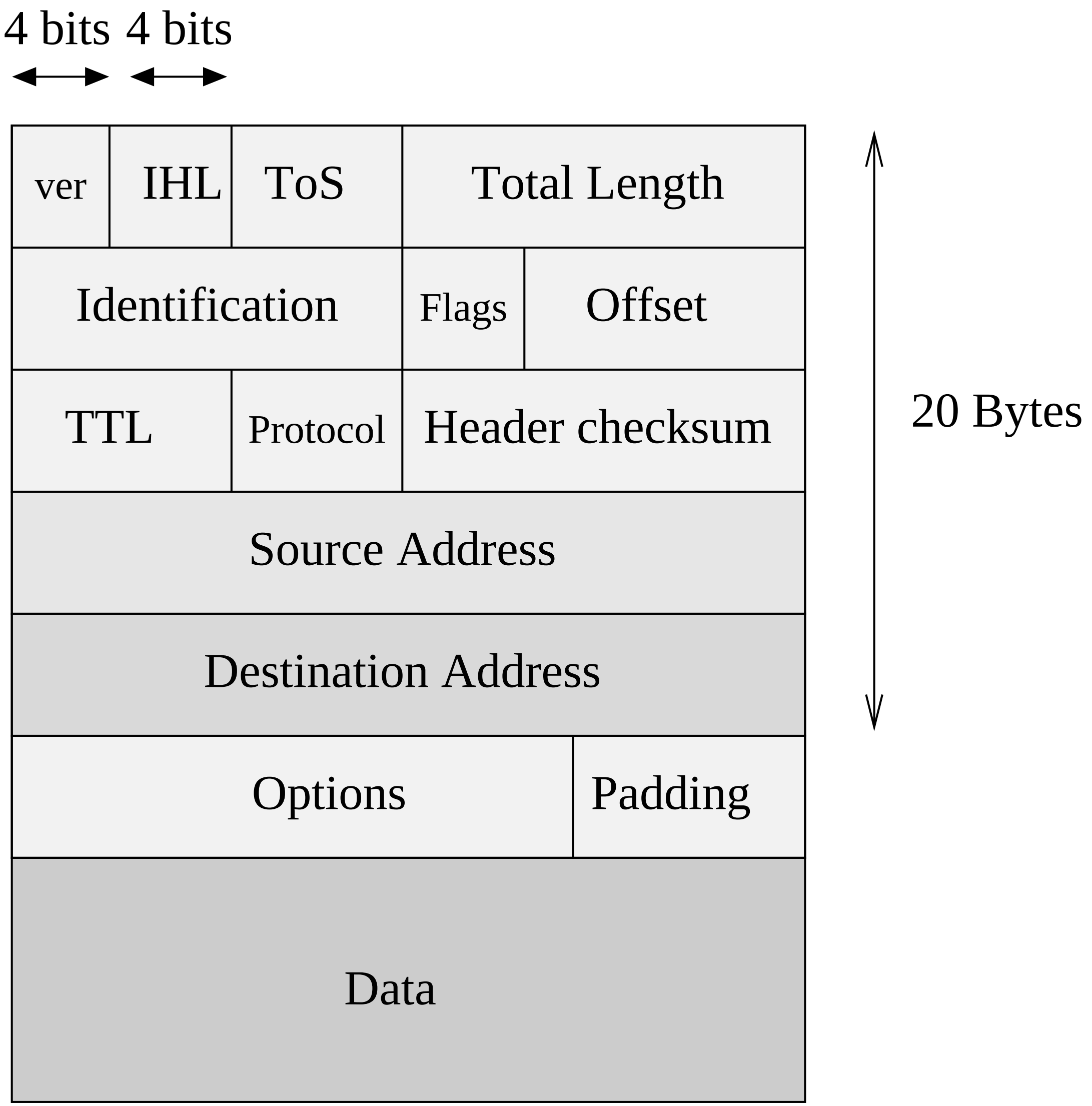
- define traffic classes and actions (policies) to block network attacks
- ACL pattern matching tool for thorough and customized packet filters
- provides match on arbitrary bits of a packet at arbitrary depth
- matches packet header + first 256 bytes of payload
- FPM provides a flexible layer 2-7 stateless classification mechanism

❖ Features

- works with IP, TCP, UDP and custom protocols defined by PHDF
- PHDF – Protocol Header Definition File (written in XML)
- pattern matching on protocol fields (eq, neq, gt, lt, value, range, regex)

3.2 Flexible Packet Matching (FPM) – Deployment

❖ Protocol Header Description File (PHDF) – example IPv4:



4 bits 4 bits

ver | IHL | ToS | Total Length

Identification | Flags | Offset

TTL | Protocol | Header checksum

Source Address

Destination Address

Options | Padding

Data

20 Bytes

IP v4

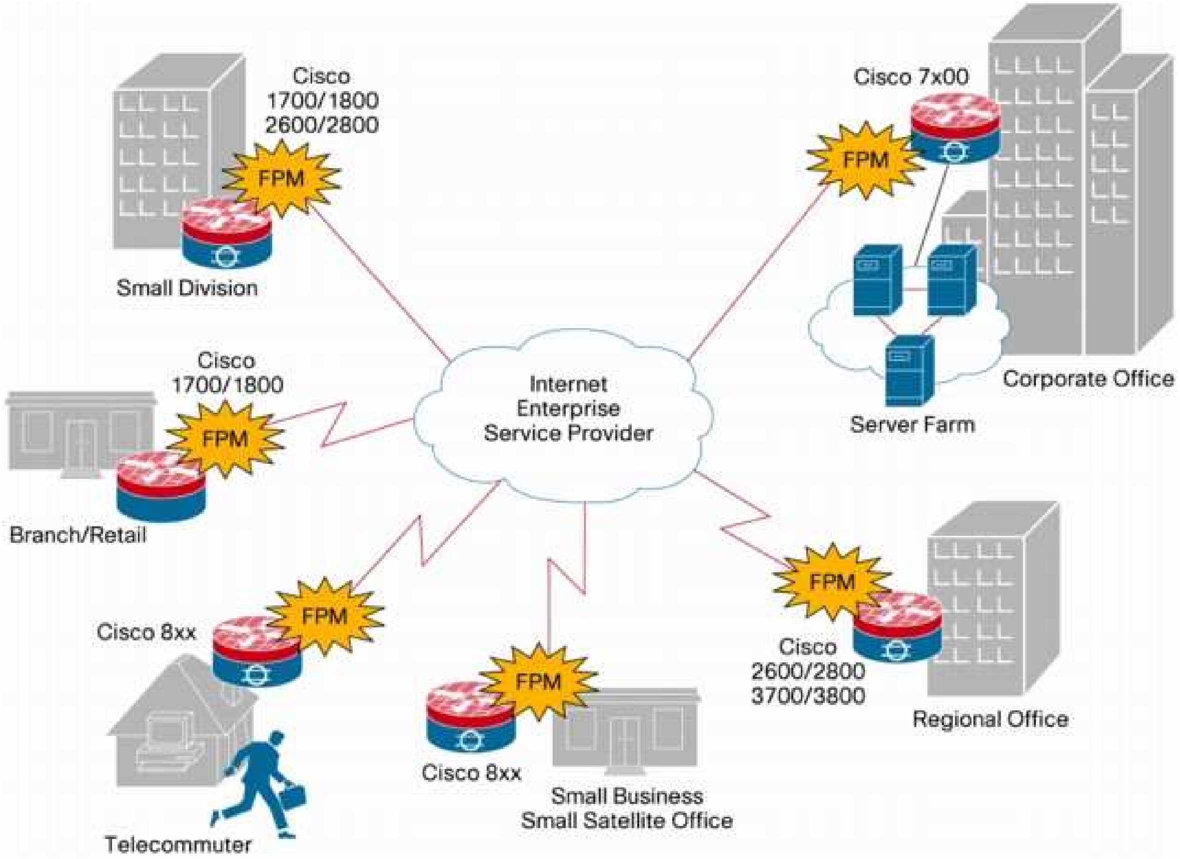
3.2 Flexible Packet Matching (FPM) – Deployment

❖ Protocol Header Description File (PHDF) – example IPv4:

```
<?xml version="1.0" encoding="UTF-8"?>
<phdf>
    <version>1</version>
        <protocol name="ip" description="Definition-for-the-IP-protocol">
            <field name="version" description="IP-version">
                <offset type="fixed-offset" units="bits">0</offset>
                <length type="fixed" units="bits">4</length>
            </field>
            <field name="ihl" description="IP-Header-Length">
                <offset type="fixed-offset" units="bits">4</offset>
                <length type="fixed" units="bits">4</length>
            </field>
            ...
            <headerlength type="fixed" value="20"></headerlength>
            <constraint field="version" value="4" operator="eq"></constraint>
            <constraint field="ihl" value="5" operator="eq"></constraint>
        </protocol>
</phdf>
```

3.2 Flexible Packet Matching (FPM) – Deployment

1. **Determine the characteristics of an attack.**
2. **Select appropriate PHDF. If does not exist, create a custom PHDF.**
3. **Load all PHDFs needed, configure class/policy maps to take an action.**
4. **Apply the service policies to appropriate interface.**



3.2 Flexible Packet Matching (FPM) – Deployment

❖ Fragmented UDP Attack

```
router(config)#load protocol flash:ip.phdf // load protocol definition

router(config)#class-map type stack match-all ip_udp // protocols to match
router(config-cmap)#description “match UDP over IP packets”
router(config-cmap)#match field ip protocol eq 0x11 next udp

router(config)#class-map type access-control match-any fragudp // patterns
router(config-cmap)#description “match on fragmented udp packets”
router(config-cmap)#match field ip flags eq 1 mask 6 // more fragment bit
router(config-cmap)#match field ip fragment-offset gt 0 // offset > 0

router(config)# policy-map type access-control fpm_frag_udp_policy /action
router(config-pmap)# description "policy for fragmented UDP based attacks"
router(config-pmap)# class fragudp
router(config-pmap-c)# drop
...
router(config)# interface GigabitEthernet 0/1 // apply on the interface
router(config-if)# service-policy type access-control input fpm_policy
```

3.2 Flexible Packet Matching (FPM) – Deployment

❖ Traffic Classification Definition File (TCDF)

- **a configuration file**
- **controls Flexible Packet Matching (FPM) features**
- **FPM uses a TCDF to define traffic classes and policies**
- **written in XML**
- **an alternative to CLI (Command Line Interface)**

3.2 Flexible Packet Matching (FPM) – Deployment

❖ TCDF for Slammer Packets:

```
<?xml version="1.0" encoding="UTF-8"?>
<tcdf>
    <class name="ip-udp" type="stack"> // define the traffic class
        <match><eq field="ip.protocol" value="0x11" next="udp"></eq></match>
    </class> // define matching criteria
    <class name="slammer" type="access-control" match="all">
        <match>
            <eq field="udp.dest-port" value="0x59A"></eq> // dest.  port 1434
            <eq field="ip.length" value="0x194"></eq> // length < 404
            // matching pattern 0x00401010 at 224 B from start of the IP headers
            <eq start="l3-start" offset="224" size="4" value="0x00401010"></eq>
        </match>
    </class>
    <policy type="access-control" name="fpm-udp-policy"> // define action
        <class name="slammer"></class>
        <action>Drop</action>
    </policy>
</tcdf>
```

3.2 Flexible Packet Matching (FPM) – Deployment

❖ Process Utilization for FPM

- run o Cisco 7206VXR Router with NPE-400 processor, 128 NB, IOS 12.4(4)T
- tests used configuration with 10 FPM classes
- 50% of 10 traffic streams generated matches on the 1st,5th, or 10th match statement
- STD – a standard IP source address match
- EXT – IP source, IP dest., TCP source port, TCP dest. port, TCP protocol match
- ALL – IP source, IP dest., TCP source port range, TCP dest., TCP SYN flag

Filter type	1000 pps	2000 pps	3000 pps	4000 pps	5000 pps
FPM STD-1-Match	16 %	33 %	49 %	64 %	70 %
FPM STD-5-Match	17 %	33 %	52 %	68 %	79 %
FPM STD-10-Match	18 %	37%	56 %	72 %	86 %
FPM EXT-1-Match	38 %	42 %	43 %	43 %	43 %
FPM EXT-5-Match	42 %	50 %	59 %	59 %	59 %
FPM EXT-10-Match	42 %	50%	50 %	50 %	50 %
FPM ALL-1-Match	51 %	30 %	50 %	50 %	50 %

3.2 Flexible Packet Matching (FPM) – Deployment

❖ Conclusion

- **FPM – pattern matching technique on a packet**
- **flexible description of the protocol and attacks**
- **stateless system**
- **a part of router's operating system IOS**
- **defines actions over attacks**
- **new protocols/attacks can be added**
- **current threats/attacks can be modified/updated**

3.3 Cisco Secure Monitoring, Analysis and Response System

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Introduction

- an applianced-based solution
- a security threat mitigation (STM) system
- identify, isolate and recommend removal of offending elements
- correlate network anomalies and security events

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Introduction

- an applianced-based solution
- a security threat mitigation (STM) system
- identify, isolate and recommend removal of offending elements
- correlate network anomalies and security events

❖ How MARS works

- processes raw events from reporting devices and sessionezes them
- analyses them and evaluates for matching inspection rules
- identifies false positives
- reduces the amount of raw data that requries manual review
- presents comprehensive view of the network

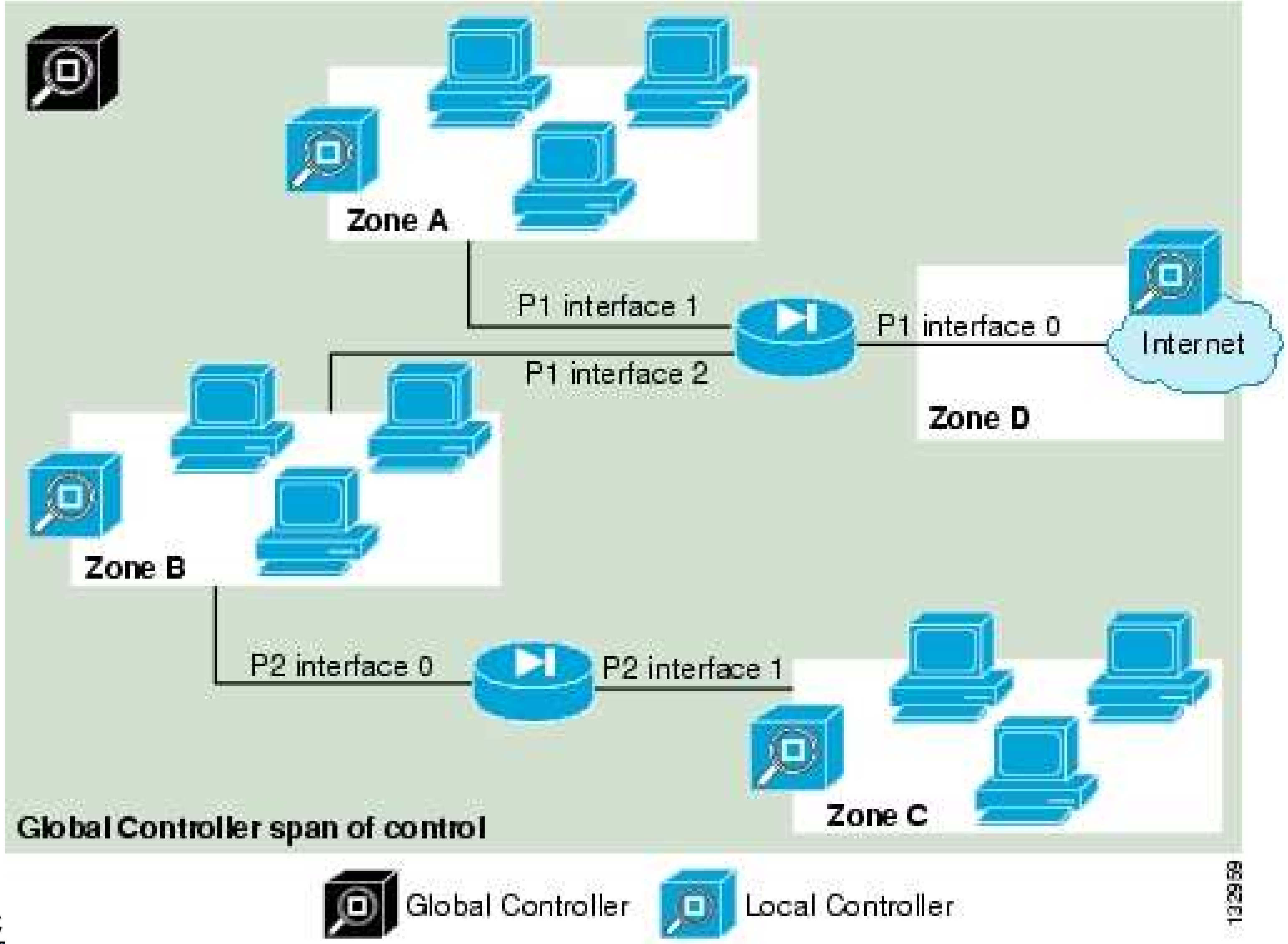
3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Components of the system

- Local Controller
  - receives and pulls data from reporting devices
  - from firewalls, routers, IDS/IPS, etc.
  - suggests mitigation rules for detected attacks
- Global Controller
  - summarizes findings of Local Controllers
  - defines new device types, inspection rules, queries
  - distributes them to Local Controllers
- MARS Web Interface
- Reporting and Mitigation Devices

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Components of the system



3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Global Data Collection in MARS – Sources:

- Dynamic vulnerability scanning
- NetFlow data collection
- L3 topology discovery
  - determine the attack path vector
  - populates the Topology graphs
- L2 device discovery
  - determine the attack path vector
  - identify attacking hosts and targets by MACs

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Global Data Collection in MARS – Sources:

- **Distributed Threat Mitigation (DTM) devices**
  - **DTM polls IPS/IDS devices to determine the top firing signatures**
  - **MARS generates the list of top signatures**
  - **IOS routers running DTM asks MARS for that list**
- **Windows event logs (every 5 mins)**
- **Oracle event logs (every 5 mins)**
- **Monitored device update scheduler**

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Reporting and mitigating devices

- Router
  - hostname, static router, ACL rules, static NAT rules
  - trafic flows, NetFlow data, ARP cache table
  - device status, resource utilization (CPU, memory, port stats)
  - Cisco router, ExtremeWare
- Switch
  - switching table, device status, NetFlow data
  - 802.1x log
  - Cisco Switch (IOS, CatOS)

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Reporting and mitigating devices

- **Firewall**
  - **interface configuration, NAT/PAT mapping, firewall policies**
  - **firewall logs, audit logs, arp cache table**
  - **Cisco PIX, ASA, Juniper Netscreen, Checpoint Opsec, Nokia Firewall**
- **VPN**
  - **remote user info, login/logout records, device status**
  - **Cisco VPN Concentrator**
- **Network IDS/IPS**
  - **fired signature alerts, trigger packet info**
  - **Cisco NIDS, NIPS, IPS ASA, IOS IPS, McAfee Intrushield**
  - **Juniper Netscreen, ISS RealSecure, Snort, CSA**

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Reporting and mitigating devices

- Host IDS, OS, Anti-Virus
  - security event logs, system logs, infected hosts
  - Windows, Solaris, Redhat
- Web servers, Web proxy, Database
  - logs via syslog
  - MS IIS, Sun iPlanet, Apache, NetApp NetCache, Oracle
- Syslog, SNMP
  - logs and traps

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ System performance

- high level of event traffic (10,000 events per second)
- 300,000 NetFlow events per second
- high-performance correlation made through inline processing logic
- embedded Oracle system

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Specification

- **Dynamic Session-Based Correlation**
  - **Anomaly detection, including Cisco NetFlow**
  - **Behaviour-based and rules-based event correlation**
  - **Automated NAT normalization**
- **Topology Discovery**
- **Vulnerability Analysis**
  - **Switch, router, firewall, and NAT configuration analysis**
  - **Incident-triggered targeted network-based and host-based fingerprinting**
  - **Automated vulnerability scanner data capture**

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Specification

- Incident Analysis and Response
  - Event management dashboard
  - Session-based event consolidation with full-rule context
  - Graphical attack path visualization
  - Attack path device profiles
  - Notification: email, pager, syslog, SNMP

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ MARS Appliances

Model	Managed Routers	Events/sec	NetFlows/sec	Storage
MARS 20R	5 devices	50	1,500	120 GB (non-RAID)
MARS 20	25 devices	500	15,000	120 GB (non-RAID)
MARS 50	25 devices	1000	30,000	240 GB RAID 0
MARS 100e	100 devices	3000	75,000	790 GB RAID 10
MARS 100	100 devices	5000	150,000	750 GB RAID 10
MARS 200	100 devices	10,000	300,000	1 TB RAID 10

❖ System specification – MARS 210, GC2R

- processor Dual Intel Woodcrest Xeon 3.0 GHz
- memory 8GB DDR2 SDRAM, Front Side Bus 1333 MHz
- PCI NIC Dual Port Intel Pro/1000 PT
- hard drive 2.0 TB-RAID 10, 6x750 GB SATA-IO HDD

3.3 Cisco Secure Monitoring, Analysis and Response System

❖ Conclusion

- monitoring, analysis and response system, not IPS
- solution for large networks
- complex solution for network diagnoses and protection
- gets data (configs, alarms, logs) from different network devices
- correlate incidents
- useful for DDoS protection
- global view on the network – devices share info about attacks
- advanced configuration requries skilled admins
- work mostly with Cisco devices (routers, firewalls etc.)

3.4 Cisco Self-Defending Network (SDN)

3.4 Cisco Self-Defending Network (SDN)

❖ Complex network protection

- using combination of different techniques, and
- combination of network active and passive devices

3.4 Cisco Self-Defending Network (SDN)

❖ **Complex network protection**

- using combination of different techniques, and
- combination of network active and passive devices

❖ **Critical components of network security**

- **Secure Network Platform** – firewall, IPSec, VPNs, SSLs, IPSs, NAC
- **Confidential Communication** – SSLs, VPNs
- **Secure Transactions** – application-layer security
- **Threat Control and Containment** – HIPS/NIPS, CSA, AV protection
- **Operational Management and Policy Control** – MARS

3.4 Cisco Self-Defending Network (SDN)

❖ Building blocks of SDN

- Secure data transmission
- End hosts protection
- Access control, infection containtment
- Intrusion detection, anomaly detection
- Intelligent monitoring
- Securing applications

3.4 Cisco Self-Defending Network (SDN)

❖ False alarms mitigation

- **Event Severity + Signature Fidelity + Attack Relevance + Asset Value of Targe ⇒ Risk Rating**

Risk Rating Threshold	Action
$0 < RR < 35$	**Alarm**
$35 < RR < 85$	**Alarm & Log Packet**
$85 < RR < 100$	**Drop packet**

3.4 Cisco Self-Defending Network (SDN)

❖ False alarms mitigation

- **Event Severity + Signature Fidelity + Attack Relevance + Asset Value of Targe ⇒ Risk Rating**

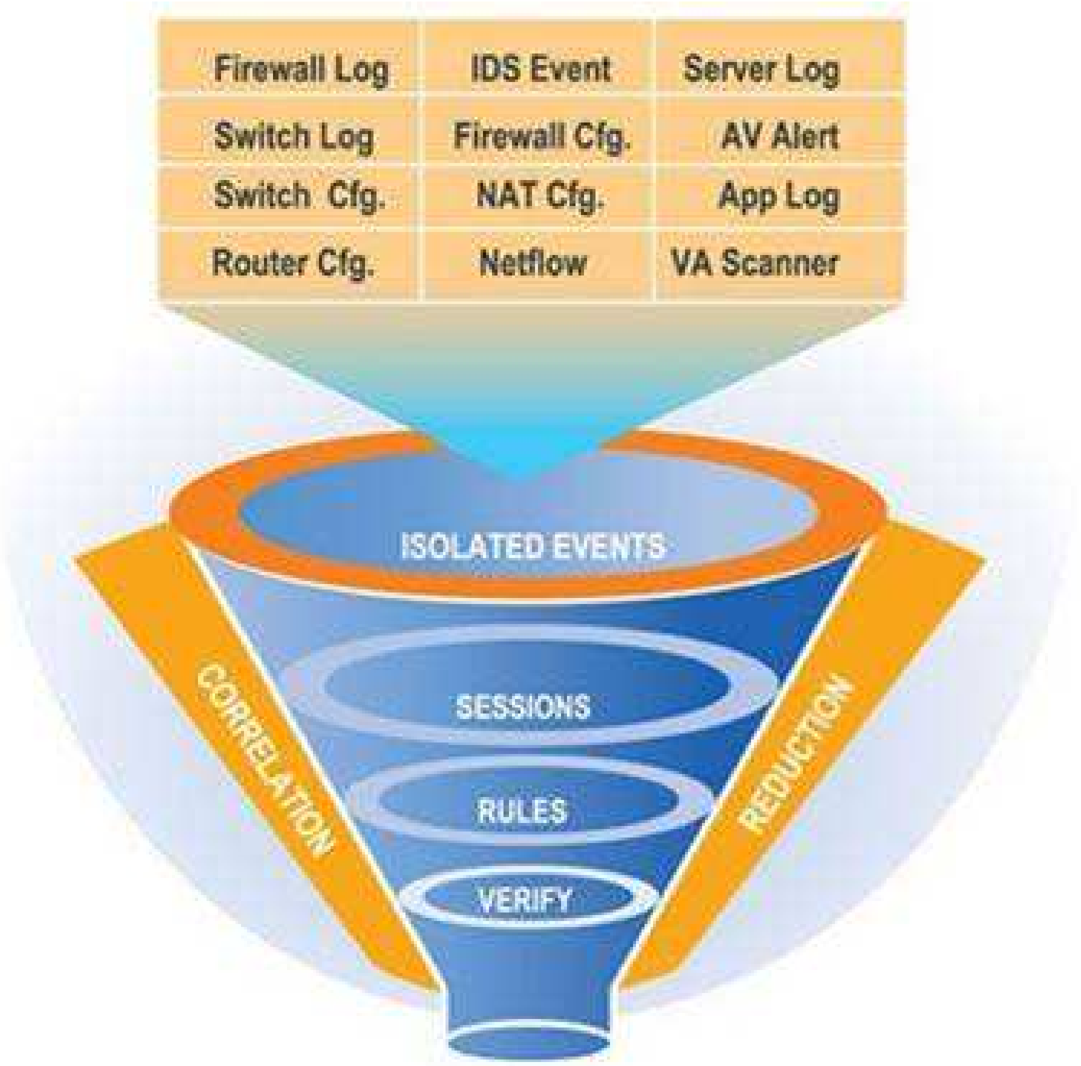
Risk Rating Threshold	Action
$0 < RR < 35$	Alarm
$35 < RR < 85$	Alarm & Log Packet
$85 < RR < 100$	Drop packet

❖ Intelligent Correlation and Incident Response

- **overlaying feedback from a variety of points ⇒ firewalls, NIDS, routers, switches, hosts**
- **learning about L2 and L3 topology**
- **attack visualization and tracing**

3.4 Cisco Self-Defending Network (SDN)

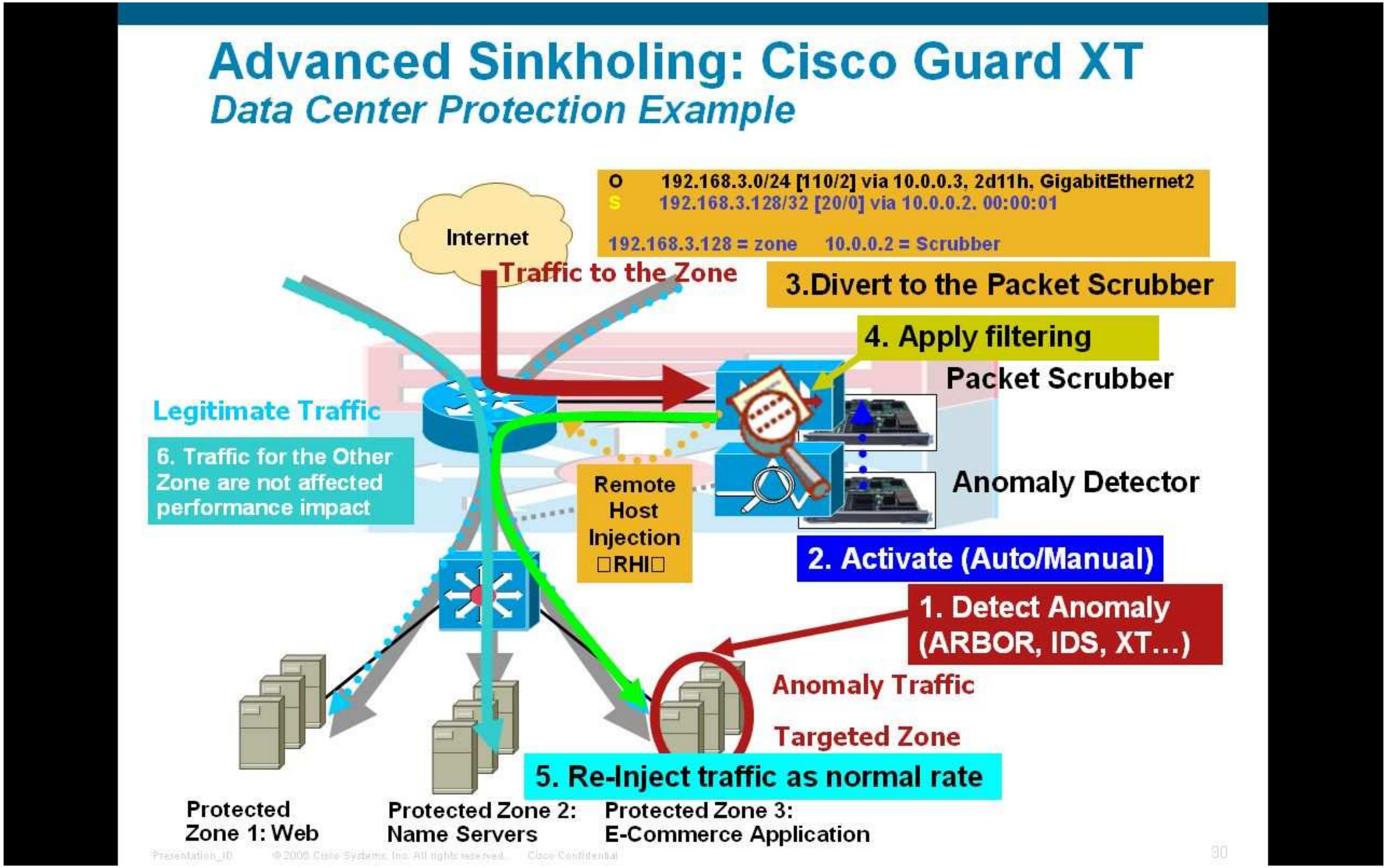
❖ Intelligent Correlation and Incident Response



Firewall Log	IDS Event	Server Log
Switch Log	Firewall Cfg.	AV Alert
Switch Cfg.	NAT Cfg.	App Log
Router Cfg.	Netflow	VA Scanner

3.4 Cisco Self-Defending Network (SDN)

❖ Sinkhole protection



Advanced Sinkholing: Cisco Guard XT

Data Center Protection Example

```
O     192.168.3.0/24 [110/2] via 10.0.0.3, 2d11h, GigabitEthernet2
S     192.168.3.128/32 [20/0] via 10.0.0.2. 00:00:01

192.168.3.128 = zone     10.0.0.2 = Scrubber
```

Internet
Traffic to the Zone
3.Divert to the Packet Scrubber
4. Apply filtering
Packet Scrubber
Legitimate Traffic
6. Traffic for the Other Zone are not affected performance impact
Remote Host Injection □RHI□
Anomaly Detector
2. Activate (Auto/Manual)
1. Detect Anomaly (ARBOR, IDS, XT…)
Anomaly Traffic
Targeted Zone
5. Re-Inject traffic as normal rate
Protected Zone 1: Web
Protected Zone 2: Name Servers
Protected Zone 3: E-Commerce Application
Presentation_ID © 2006 Cisco Systems, Inc. All rights reserved. Cisco Confidential
30

3.4 Cisco Self-Defending Network (SDN)

❖ Incident Dashboard

- aggregate
- correlate
- summarize

❖ Incident Filtering

2,694.083 events → 992.511 sessions → 249 incidents → 61 high severity incidents

4 Advanced Techniques for Traffic Analysis

4 Advanced Techniques for Traffic Analysis

❖ Cisco Service Control Engine

- session classification
- control of application-level IP traffic per subscriber
- deep packet inspection

❖ ModSecurity

- HTTP security

4.1 Cisco Service Control Engine (SCE)

4.1 Cisco Service Control Engine (SCE)

❖ Cisco SCE Introduction

- **a purpose-build hardware device for service providers**
- **classification, analysis and control of Internet/IP traffic**
- **ISP can analyse, charge for, and control IP traffic at multigigabit speeds**

4.1 Cisco Service Control Engine (SCE)

❖ Cisco SCE Introduction

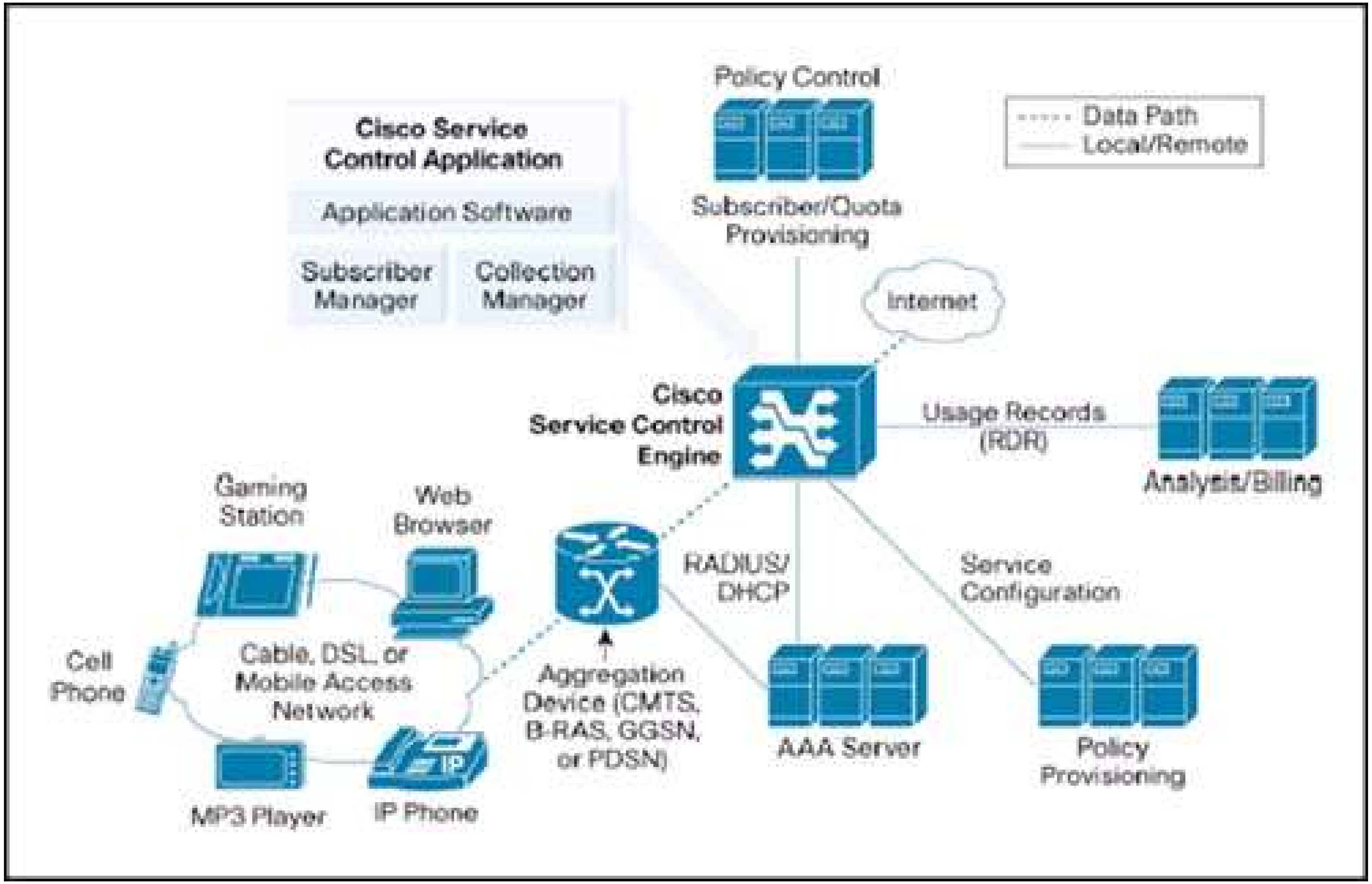
- a purpose-build hardware device for service providers
- classification, analysis and control of Internet/IP traffic
- ISP can analyse, charge for, and control IP traffic at multigigabit speeds

❖ SCE features

- session-based classification
- control of application-level IP traffic per subscriber
- deep packet inspection for multi-gigabit and 10 gigabit speeds
- reconstructs flows at the Layer 7 state of each application flow
- programmable and extensible through Service Management Language (SML)

4.1 Cisco Service Control Engine (SCE)

❖ Deployment



4.1 Cisco Service Control Engine (SCE)

❖ The core of the Service Control Engine

- Application-layer stateful-flow inspection of IP traffic
- Using ASIC components and RISC processors
- Robust support for over 600 protocols and applications:
  - General: http, https, ftp, telnet, nntp, smtp, pop3, imap, wap
  - P2P file sharing: FastTrack-KazaA, Gnutella, BitTorrent
  - P2P VoIP: Skype, Skinny, DingoTel
  - Multimedia: RTSP, SIP, HTTP streaming, RTP/RTCP
- programmable system core for flexible reporting and bandwidth control

4.1 Cisco Service Control Engine (SCE)

❖ SCE Management and Collection

- Network Management
  - Faults, Configuration, Accounting, Performance, Security
- Subscriber Management
  - different policies on different subscribers
  - mapping network IDs to subscriber IDs
  - combination of DHCP, AAA, Radius services

4.1 Cisco Service Control Engine (SCE)

❖ SCE Management and Collection

- Network Management
  - Faults, Configuration, Accounting, Performance, Security
- Subscriber Management
  - different policies on different subscribers
  - mapping network IDs to subscriber IDs
  - combination of DHCP, AAA, Radius services
- Service Configuration Management
  - definition of service control application
  - traffic classification, accounting, reporting
- Data Collection
  - data and statistics in Raw Data Records (RDR) format
  - Collection Manager (CM) listens on RDRs and process them

4.1 Cisco Service Control Engine (SCE)

❖ Conclusion

- combination of a special hardware device and software solution
- provides traffic analysis and classification
- collects data, make statistics and accounting reports
- application-layer stateful data inspection
- programmable solution with Service Modelling Language (SML)

4.2 HTTP Security

4.2 HTTP Security

❖ ModSecurity(tm), Breach

- **a web application firewall (WAF)**
- **provides HTTP traffic monitoring, RT analysis, attack detection**
- **works as Web IDS**
- **can be a part of the web server, or Apache-based reverse proxy server**
- **distributed under GNU GPL or commercial licences with a support**

4.2 HTTP Security

❖ ModSecurity(tm), Breach

- a web application firewall (WAF)
- provides HTTP traffic monitoring, RT analysis, attack detection
- works as Web IDS
- can be a part of the web server, or Apache-based reverse proxy server
- distributed under GNU GPL or commercial licences with a support

❖ Flexible Rule Engine

- implements ModSecurity Rule Language

4.2 HTTP Security – ModSecurity

❖ **Attack prevention**

1. **Negative security model**
   - **monitors requests for anomalies, unusual behaviour, common web attacks**
   - **keeps anomaly score for each request, IP, session and user account**
   - **requests with high anomaly scores are logged or rejected**

4.2 HTTP Security – ModSecurity

- **Attack prevention**

1. **Negative security model**
   - **monitors requests for anomalies, unusual behaviour, common web attacks**
   - **keeps anomaly score for each request, IP, session and user account**
   - **requests with high anomaly scores are logged or rejected**
2. **Positive security model**
   - **only valid requests are accepted**
   - **best for application that are heavily used but rarely updated**

4.2 HTTP Security – ModSecurity

❖ **ModSecurity Core Rules Structure includes**

- **the logic required to detect attacks**
- **a policy setting the actions to perform if an attack is detected**
- **information regarding attack**

4.2 HTTP Security – ModSecurity

❖ **ModSecurity Core Rules Structure includes**

- the logic required to detect attacks
- a policy setting the actions to perform if an attack is detected
- information regarding attack

❖ **Core Rules Content**

- HTTP protection – violation of the HTTP protocol
- Common Web Attacks Protection
- Automation detection – bots, crawlers, scanners etc.
- Trojan Protection – access to Trojans horses
- Error Hiding – Disguising error messages sent by the server

4.2 HTTP Security – ModSecurity

❖ HTTP protection

- SQL Injection
- Cross-Site Scripting
- OS Command execution
- Remote code inclusion
- LDAP Injection
- SSI Injection
- Information leak
- Buffer overflows
- File disclosure

4.2 HTTP Security – ModSecurity

❖ Example – HTTP violation

```
# Accept only digits in content length
#
SecRule REQUEST_HEADERS:Content-Length "!^\d+$"
"deny,log,auditlog,status:400,msg:’Content-Length HTTP header is not
numeric’, severity:’2’„id:’960016’,"
```

❖ Example – protocol anomalies

```
SecRule REQUEST_HEADERS:User-Agent "@eq 0" \
"skip:1, log,auditlog, msg:’Request Missing a User Agent Header’
„id:’960009’, severity:’4’"

SecRule REQUEST_HEADERS:User-Agent "^$" \
"log,auditlog,msg:’Request Missing a User Agent Header’„id:’960009’,
severity:’4’"
```

4.2 HTTP Security – ModSecurity

❖ Example – protocol policy

```
# Restrict file extension
#
# TODO the list of file extensions below are virtually always considered unsafe
#       and not in use in any valid program. If your application uses one of
#       these extensions, please remove it from the list of blocked extensions.
#       You may need to use ModSecurity Core Rule Set Templates to do
#       so, otherwise comment the whole rule.
#
SecRule REQUEST_BASENAME "\.(?:c(?:o(?:nf(?:ig)?|m)|s(?:proj|r)?|dx|er|fg|md)|
p(?:rinter|ass|db|ol|wd)|v(?:b(?:proj|s)?|sdisco)|a(?:s(?:ax?|cx)|xd)|d(?:bf?|
at|ll|os)|i(?:d[acq]|n[ci])|ba(?:[kt]|ckup)|res(?:ources|x)|s(?:h?tm|ql|ys)|
l(?:icx|nk|og)|\w{,5}~|webinfo|ht[rw]|xs[dx]|key|mdb|old)$" \
    "t:urlDecodeUni, t:lowercase, deny,log,auditlog,status:500,msg:’URL file
extension is restricted by policy’, severity:’2’,,id:’960035’,"
```

4.2 HTTP Security – ModSecurity

❖ Conclusion

- an application specific IDS
- syntactical protocol analysis
- attack detection based on signatures (regular expressions)
- flexible extension, adding new rules

Conclusion of the talk

❖ Basic Principles for Building Network Security

- complex solution required – not a single device
- both signature and anomaly based detection
- weighted correlation of different incident events, logs etc.
  - packet and flow analysis and processing
    ⇒ huge disk capacity
- analysis of high level protocols
  - a description language for protocols, attacks, response
  - simple format – adding new rules
- combination of fast hardware processing and software solution

Použitá literatura

- Fundamentals of Network Security. CiscoPress, 2002.
- An Introduction to Intrusion Detection Assesment. ICSA, Inc, 1999.
- Cisco Traffic Anomaly Detection. Cisco Web Site.
- Flexible Packet Matching XML Deployment Guide. Cisco Web Site.
- CS MARS. Cisco Web site.
- Cisco Service Control Engine Cisco Web Site
- ModSecurity Reference Manual. ModSecurity Web Site.
- Cisco Expo 2007 – Security Techtorial. Lecture notes.