Moloch

A New and Free Way To Index Your Packet Capture Repository

Andy Wick and Eoin Miller

ShmooCon IX – February 2013

Project Logo

It is a Great Horned Owl

Will There Be Lots Of Super Awesome Owl Pictures In This Presentation?

Yes. But there is no picture of an owl that isn't super awesome. Also, there may be random facts about owls. There will be a quiz later at the bar.

Introductions

Andy Wick

- Developer
- COPO (Chief Owl Procurement Officer)
- Superstar Par Excellence

Introductions

Eoin Miller

- IDS/PCAP Centric Security Nerd
- Falconer/Owl Wrangler
- Anti-Malvertising Enthusiast

Moloch – Overview – What Is Moloch?

Moloch is an open source, scalable IPv4 packet capturing (PCAP) indexing and database system.

- A simple web GUI is provided for browsing, searching, viewing and exporting PCAP data
- Web API's are accessible if you wish to design your own GUI or directly grab PCAP with various command line tools for further analysis or processing
- Download it from AOL's GitHub page: https://github.com/AOL/Moloch
- It is like AOL Search for PCAP repositories!

Moloch – Overview – What Moloch Is Not

- An IDS/IPS Engine
- Firewall/Filtering/Multifunction Device
- Expensive
- Slow
- Without appetite for disk space and memory

Moloch – What Was The Need?

- The open source community was lacking a fast, flexible method of capturing and indexing PCAP

- Commercial Off The Shelf (COTS) products that exist were very cost prohibitive especially when compared against the low cost of hardware alone

- Other open source projects existed that allowed for capture and retrieval of specific sessions from specific files based on an indicator, but none met our requirements or the feature set of most commercial offerings

Why The Owl Logo?

Owls are silent hunters that go after RAT's. We think that's pretty cool.

Moloch – Uses

- Real-time capture of network traffic for forensic and investigative purposes
  - Combine the power of Moloch with other indicators (intelligence feeds, alerting from IDS/anti-virus) to empower your Analysts to quickly and effectively review actions on the network to determine the validity/threat
  - The ability to review past network traffic for post compromise investigations
- Static PCAP repository
  - Large collections of PCAP that is created by malware
  - Collections of PCAP from various CTF events
  - Custom tagging of data at time of import
- Put it in front of your sinkhole, honeypot or darknet

Moloch – Components

- Capture
  - A single-threaded C application that sniffs the network interface, parses the traffic and creates the Session Profile Information (aka SPI-Data) and writes the packets to disk
- Database
  - For storing and searching through the SPI-Data generated by the capture component
- Viewer
  - A web interface that allows for GUI and API access from remote hosts to browse/query SPI-Data and retrieve stored PCAP

Moloch – Components – Capture

- Single threaded libnids based daemon written in C
- Can be used to sniff network interface for live capture to disk
- Can be called from the command line to do manual import of PCAP for parsing and storage
- Parses various layer 3-7 protocols, creates “session profile information” aka SPI-Data and spits them out to the elasticsearch cluster for indexing purposes
- Kind of like making owl pellets!

Moloch – Components – Database

- elasticsearch (http://www.elasticsearch.org)
  - Powered by Apache's Lucene (http://lucene.apache.org)
  - Requests received in URI's over HTTP
  - Results returned in JSON
- nosql
- Document oriented (which is great for lots and lots of network sessions)
- Automatic sharding across multiple hosts by magic elves
- Fast, scalable, all that goodness

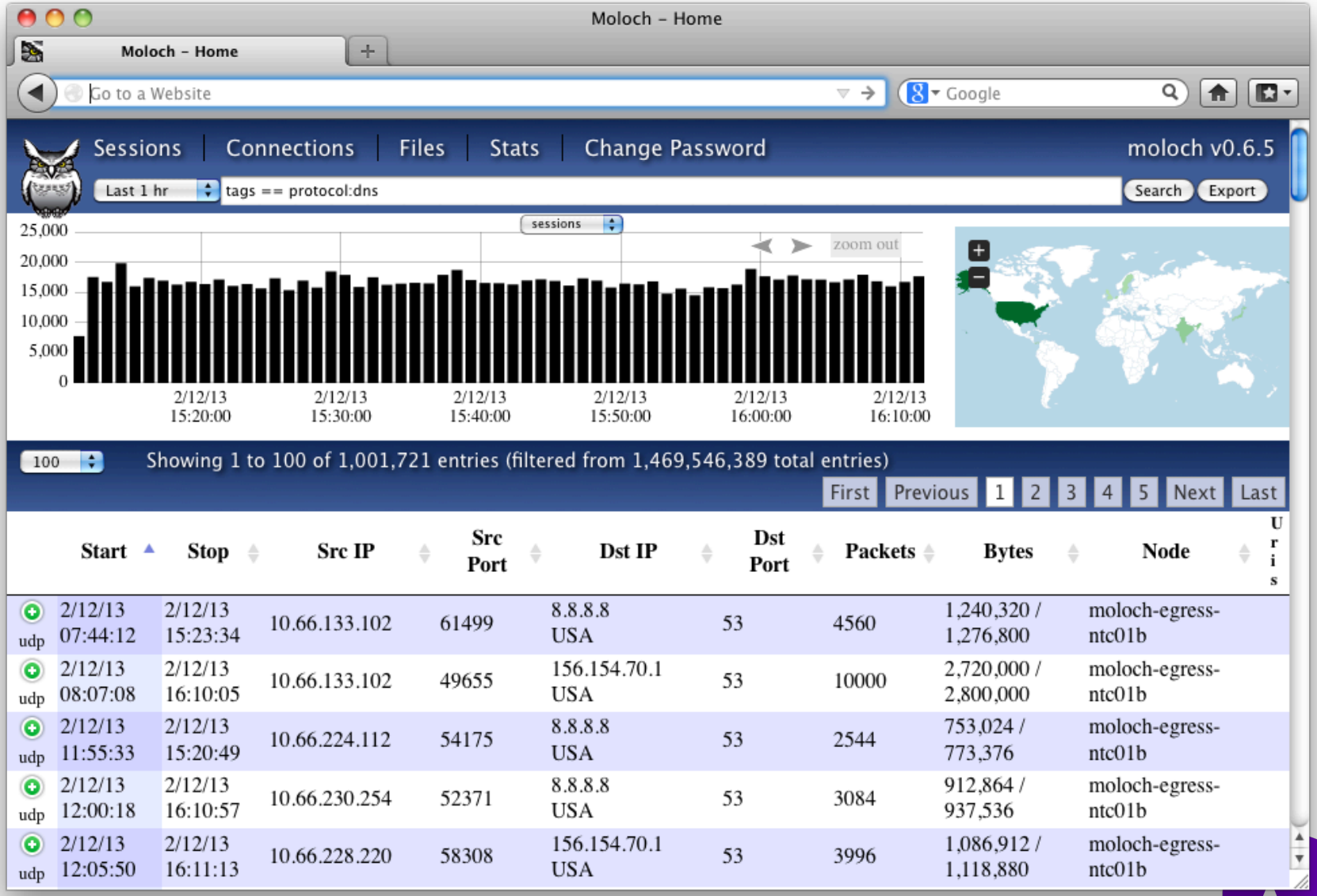
Moloch – Components – Viewer

- nodejs based application
  - nodejs is an event driven server side JavaScript platform based on Google Chrome's JavaScript runtime
  - Comes with its own HTTP server and likes it some JSON for communication
  - http://nodejs.org - server side JavaScript is for the cool kids!

- Provides web based GUI for browsing/searching/viewing/ exporting SPI-data and PCAP

- GUI/API's calls are all done through URI's so integration with SEIM's, consoles and command line tools is easy for retrieving PCAP or sessions of interest

Moloch – Components – Viewer



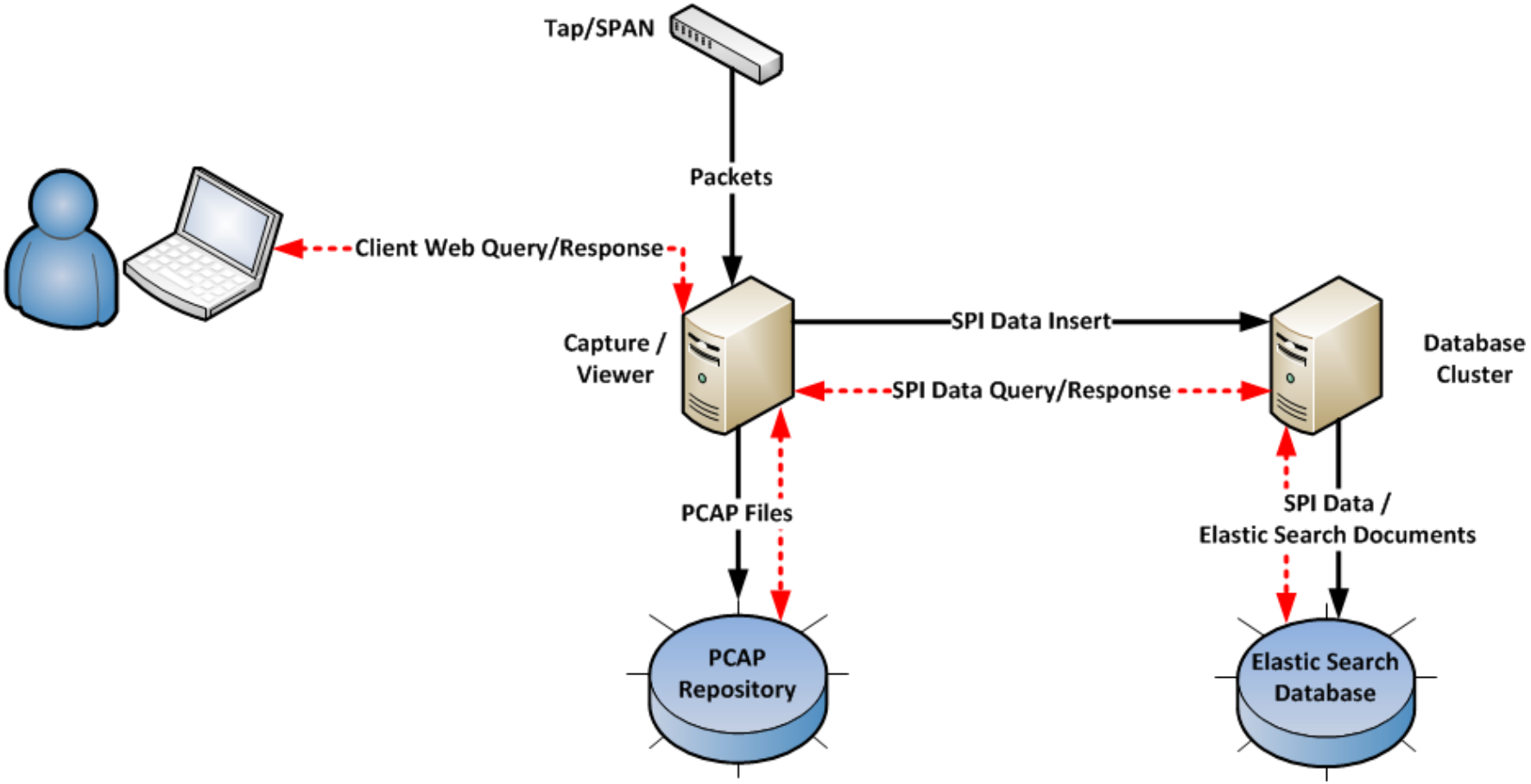
	Start	Stop	Src IP	Src Port	Dst IP	Dst Port	Packets	Bytes	Node	Uris
udp	2/12/13 07:44:12	2/12/13 15:23:34	10.66.133.102	61499	8.8.8.8 USA	53	4560	1,240,320 / 1,276,800	moloch-egress-ntc01b	
udp	2/12/13 08:07:08	2/12/13 16:10:05	10.66.133.102	49655	156.154.70.1 USA	53	10000	2,720,000 / 2,800,000	moloch-egress-ntc01b	
udp	2/12/13 11:55:33	2/12/13 15:20:49	10.66.224.112	54175	8.8.8.8 USA	53	2544	753,024 / 773,376	moloch-egress-ntc01b	
udp	2/12/13 12:00:18	2/12/13 16:10:57	10.66.230.254	52371	8.8.8.8 USA	53	3084	912,864 / 937,536	moloch-egress-ntc01b	
udp	2/12/13 12:05:50	2/12/13 16:11:13	10.66.228.220	58308	156.154.70.1 USA	53	3996	1,086,912 / 1,118,880	moloch-egress-ntc01b	

Moloch – Architecture

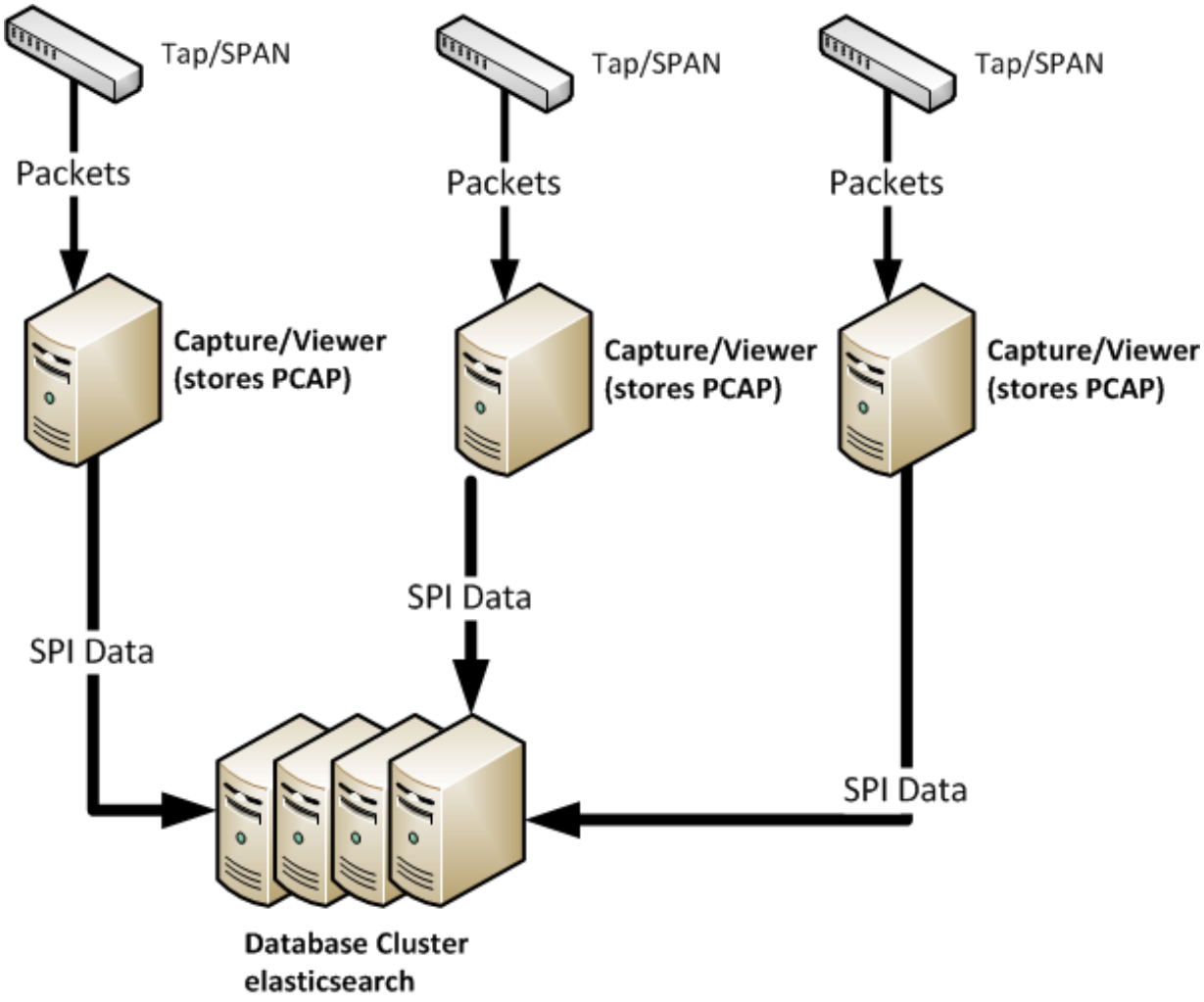
- All components (Capture, Database and Viewer) can exist and operate on the same host
  - Capture will want lots of storage space for PCAP that has been ingested
  - Database will want lots of RAM for indexing and fast searching
  - Viewer is very small and can go anywhere really
  - Not recommended for large amounts of PCAP throughput

- Can scale easily across multiple hosts for Capture and Database components easily
  - One or more Capture instances can run on one or more hosts and report to the Database
  - Database can run on one or more hosts to expand amount of RAM available for indexing
  - Best setup for capturing and indexing live traffic for investigations and defending your network

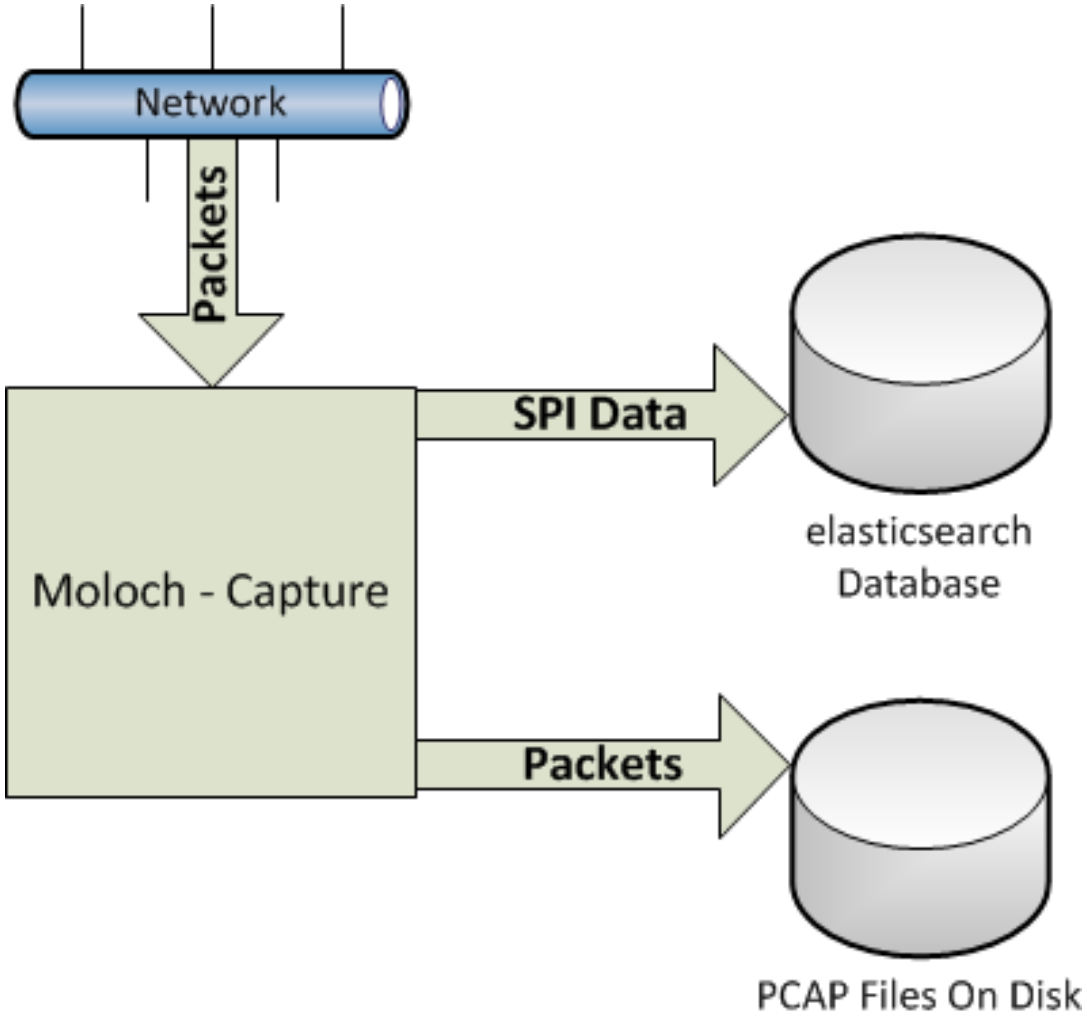
Moloch – Architecture – Overall Data Flow



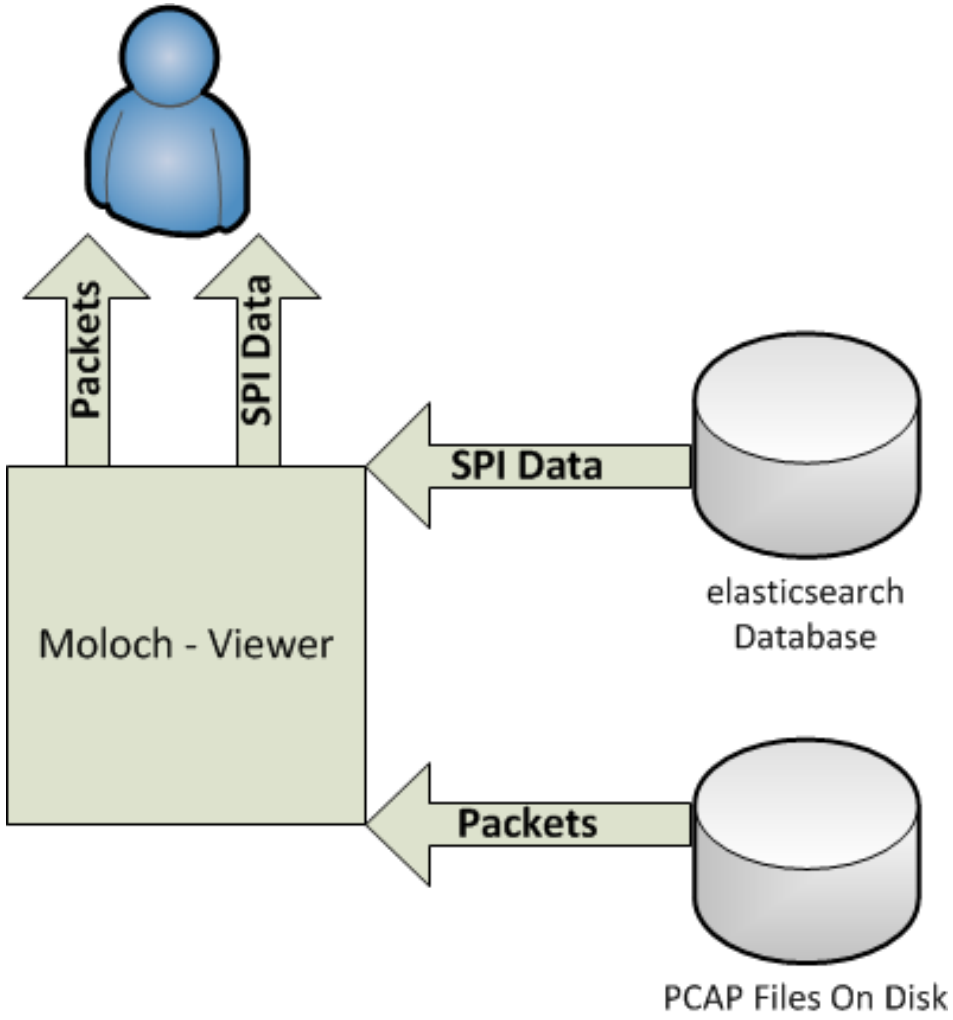
Moloch – Architecture – MultiNode w/Cluster



Moloch – Architecture – Input Data Flow



Moloch – Architecture – Output Data Flow



Owls Are Immortal

This owl was hit head on by a Ford Expedition doing 60 miles an hour. It just sat in the radiator for for the rest of the hour long journey plus another two days eating bugs. Not even a broken bone.

Moloch – Capture – SPI-Data Types

- Moloch parses various protocols to create SPI-Data:
  - IP
  - HTTP
  - DNS
    - IP Address
    - Hostname
  - SSH
    - Client Name
    - Public Key
  - SSL/TLS
    - Certificate elements of various types (common names, serial, etc)

- This is not an all inclusive list

Moloch – Capture – Creating SPI-Data

```
Start Time: 2/13/13 21:43:56
Stop Time : 2/13/13 21:44:04
Databytes/Bytes: 9,315/14,288
IP Protocol: 6
IP/Port: 172.128.1.1:52465 (USA) [AS1668 AOL Transit Data Network]
         205.188.18.208:80 (USA) [AS1668 AOL Transit Data Network]

Tags: http:content:application/octet-stream http:method:GET
http:statuscode:200 node:egress node:moloch-egress-dtc01 protocol:http tcp

Request Headers:accept accept-encoding accept-language connection cookie host user-agent
Response Headers:accept-ranges connection content-length content-type date keep-alive
server set-cookie

User Agents:'Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101 Firefox/16.0'
```

- TCP session transcript

```
GET /favicon.ico?v=2 HTTP/1.1
Host: www.aol.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101
Firefox/16.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cookie: <REDACTED>

HTTP/1.1 200 OK
Date: Wed, 13 Feb 2013 21:43:57 GMT
Server: Apache
Set-Cookie: RSP_CHECK_PORTAL_STARTPAGE.AOL.COM=deleted; expires=Thu Jan
01 00:17:51 1970 GMT; path=/; domain=www.aol.com
Accept-Ranges: bytes
Content-Length: 7886
Keep-Alive: timeout=5, max=71
Connection: Keep-Alive
Content-Type: image/x-icon
```

Moloch – Capture – Creating SPI-Data

```
Start Time: 2/13/13 21:43:56
Stop Time : 2/13/13 21:44:04
Databytes/Bytes: 9,315/14,288
IP Protocol: 6
IP/Port: 172.128.1.1:52465 (USA) [AS1668 AOL Transit Data Network]
         205.188.18.208:80 (USA) [AS1668 AOL Transit Data Network]

Tags: http:content:application/octet-stream http:method:GET
http:statuscode:200 node:egress node:moloch-egress-dtc01 protocol:http tcp

Request Headers:accept accept-encoding accept-language connection cookie host user-agent
Response Headers:accept-ranges connection content-length content-type date keep-alive
server set-cookie

User Agents:'Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101 Firefox/16.0'
Hosts:www.aol.com
URI: www.aol.com/favicon.ico?v=2
```

- All Session Profile Information (SPI-Data) Created

```
User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101
Firefox/16.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cookie: <REDACTED>

HTTP/1.1 200 OK
Date: Wed, 13 Feb 2013 21:43:57 GMT
Server: Apache
Set-Cookie: RSP_CHECK_PORTAL_STARTPAGE.AOL.COM=deleted; expires=Thu Jan
01 00:17:51 1970 GMT; path=/; domain=www.aol.com
Accept-Ranges: bytes
Content-Length: 7886
Keep-Alive: timeout=5, max=71
Connection: Keep-Alive
Content-Type: image/x-icon
```

Moloch – Capture – Creating SPI-Data

```
Start Time: 2/13/13 21:43:56
Stop Time : 2/13/13 21:44:04
Databytes/Bytes: 9,315/14,288
IP Protocol: 6
IP/Port: 172.128.1.1:52465 (USA) [AS1668 AOL Transit Data Network]
         205.188.18.208:80 (USA) [AS1668 AOL Transit Data Network]

Tags: http:content:application/octet-stream http:method:GET
http:statuscode:200 node:egress node:moloch-egress-dtc01 protocol:http tcp
```

- Based off of the TCP session data
- Session Start/End Timestamps
- Databytes == total number of bytes in the payload of all packets in the session
- Bytes == total number of bytes in the session, includes headers and payload
- IP Protocol == Protocol number (6 == TCP)
- IP address of source/destination.
- Port of source/destination
- Country of source/destination
- ASN of source/destination IP address

```
Cookie: <REDACTED>

HTTP/1.1 200 OK
Date: Wed, 13 Feb 2013 21:43:57 GMT
Server: Apache
Set-Cookie: RSP_CHECK_PORTAL_STARTPAGE.AOL.COM=deleted; expires=Thu Jan
01 00:17:51 1970 GMT; path=/; domain=www.aol.com
Accept-Ranges: bytes
Content-Length: 7886
Keep-Alive: timeout=5, max=71
Connection: Keep-Alive
Content-Type: image/x-icon
```

Moloch – Capture – Creating SPI-Data

```
Start Time: 2/13/13 21:43:56
Stop Time : 2/13/13 21:44:04
Databytes/Bytes: 9,315/14,288
IP Protocol: 6
IP/Port: 172.128.1.1:52465 (USA) [AS1668 AOL Transit Data Network]
         205.188.18.208:80 (USA) [AS1668 AOL Transit Data Network]

Tags: http:content:application/octet-stream http:method:GET
http:statuscode:200 node:egress node:moloch-egress-dtc01 protocol:http tcp

Requ
Resp
serv

User
Host
URI:

GET
Host
User
Fire
Acce
Acce
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cookie: <REDACTED>

HTTP/1.1 200 OK
Date: Wed, 13 Feb 2013 21:43:57 GMT
Server: Apache
Set-Cookie: RSP_CHECK_PORTAL_STARTPAGE.AOL.COM=deleted; expires=Thu Jan
01 00:17:51 1970 GMT; path=/; domain=www.aol.com
Accept-Ranges: bytes
Content-Length: 7886
Keep-Alive: timeout=5, max=71
Connection: Keep-Alive
Content-Type: image/x-icon
```

- Custom tags applied to the session
- http:content:application/octet-stream == file type fingerprint
- http:method:GET == HTTP client method
- http:statuscode:200 == HTTP status code returned from server
- node:egress == used as a grouping to identify this and others as egress traffic
- node:moloch-egress-dtc01 == node name that captured the traffic
- protocol:http == session detected as http by the parsing library (port agnostic!)

Moloch – Capture – Creating SPI-Data

```
Start Time: 2/13/13 21:43:56
Stop Time : 2/13/13 21:44:04
Databytes/Bytes: 9,315/14,288
IP Protocol: 6
IP/Port: 172.128.1.1:52465 (USA) [AS1668 AOL Transit Data Network]
         205.188.18.208:80 (USA) [AS1668 AOL Transit Data Network]

Tags: http:content:application/octet-stream http:method:GET
http:statuscode:200 node:egress node:moloch-egress-dtc01 protocol:http tcp

Request Headers:accept accept-encoding accept-language connection cookie host user-agent
Response Headers:accept-ranges connection content-length content-type date keep-alive
server set-cookie
```

- Request Headers == HTTP headers in the request for the session
- Response Headers == HTTP headers in the response for the session

```
GET /favicon.ico?v=2 HTTP/1.1
Host: www.aol.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101
Firefox/16.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cookie: <REDACTED>

HTTP/1.1 200 OK
Date: Wed, 13 Feb 2013 21:43:57 GMT
Server: Apache
Set-Cookie: RSP_CHECK_PORTAL_STARTPAGE.AOL.COM=deleted; expires=Thu Jan
01 00:17:51 1970 GMT; path=/; domain=www.aol.com
Accept-Ranges: bytes
Content-Length: 7886
Keep-Alive: timeout=5, max=71
Connection: Keep-Alive
Content-Type: image/x-icon
```

Moloch – Capture – Creating SPI-Data

```
Start Time: 2/13/13 21:43:56
Stop Time : 2/13/13 21:44:04
Databytes/Bytes: 9,315/14,288
IP Protocol: 6
IP/Port: 172.128.1.1:52465 (USA) [AS1668 AOL Transit Data Network]
         205.188.18.208:80 (USA) [AS1668 AOL Transit Data Network]
```

- User Agents == User agent string seen in the request
- Hosts == hostname seen in the session
- URI == the URI seen in the session

User Agents:'Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101 Firefox/16.0'
Hosts:www.aol.com
URI: www.aol.com/favicon.ico?v=2

```
GET /favicon.ico?v=2 HTTP/1.1
Host: www.aol.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; rv:16.0) Gecko/20100101
Firefox/16.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cookie: <REDACTED>

HTTP/1.1 200 OK
Date: Wed, 13 Feb 2013 21:43:57 GMT
Server: Apache
Set-Cookie: RSP_CHECK_PORTAL_STARTPAGE.AOL.COM=deleted; expires=Thu Jan
01 00:17:51 1970 GMT; path=/; domain=www.aol.com
Accept-Ranges: bytes
Content-Length: 7886
Keep-Alive: timeout=5, max=71
Connection: Keep-Alive
Content-Type: image/x-icon
```

OK Guys, Who Had Bean Burritos For Breakfast?

I’m serious.

Moloch – Database – Indexing

Moloch will index SPI-Data in one of three ways

- Standard Indexing
- Wildcard
- Full Text Indexing

Moloch – Database – Indexing – Standard

Standard indexing is just the unique value of the SPI-Data being indexed. Used for defined numeric values like:

- Port
- IP Protocol Type (TCP/UDP/ICMP)
- Bytes/Databytes
- Packet Count

Moloch – Database – Indexing – Wildcard

Wildcard indexing is like standard indexing but you may also use astericies to indicate wildcards in the query of the SPI-Data. Types of SPI-Data indexed like this are:

- IP Address (ip == 10.0.0.* - can also do CIDR, etc)
- Hostname (host == *.aol.com)
- Header (header == *auth* - find us some auth headers! Wait until demo time!)

Moloch – Database – Indexing – Full Text

Full text indexing will index every continuous word character string within a SPI-Data element. Types of SPI-Data indexed this way are:

- ASN (asn == AOL – any ASN name that has the word AOL)
- URI (uri == login – any URI that has the word login)
  - Matches would be /login.php, /login.asp, /login.derp
  - NOT matching would be /login**s**.php, /**1**login.asp
- User Agent (ua == Java – any user agent that has the word Java)

Moloch – Database – Indexing – Full Text

Lets take a look at how Moloch would perform full text indexing on the below URI:

```
daol.aol.com/?icid=navbar_rootmore_main5
```

Moloch splits URI's up using non-word characters as delimiters. Non-word characters (delimiters) are shown below in **bold red**:

```
daol.aol.com/?icid=navbar_rootmore_main5
```

So the following URI SPI-Data type strings could be searched for to find this session:

```
daol, aol, com, icid, navbar_rootmore_main5
```

Moloch – Viewer – Searching

SPI-Data types can overlap from various sources. Moloch makes searching for sessions containing that information easy. IP addresses exist in places such as:

- IP Packet Header (ip.src, ip.dst)
- DNS Query Responses (ip.dns)
- SMTP Mail Headers (ip.email)
- HTTP X-Forwarded-For Headers (ip.xff)

Moloch lets your query all these locations by simply asking:

```
ip == 1.1.1.1
```

Moloch – Other Awesomeness

- Supports YARA rules
- Plugin Architecture
- Custom tagging based on lists of IP addresses or hosts
- Other stuff…
- Submit requests for things in github, this is actively maintained http://github.com/AOL/Moloch

Questions?

Live Demo Time!

Moloch has been running since before ShmooCon started on the network! Lets see what we can find! Lets get to hootin!

Aol.