WISE

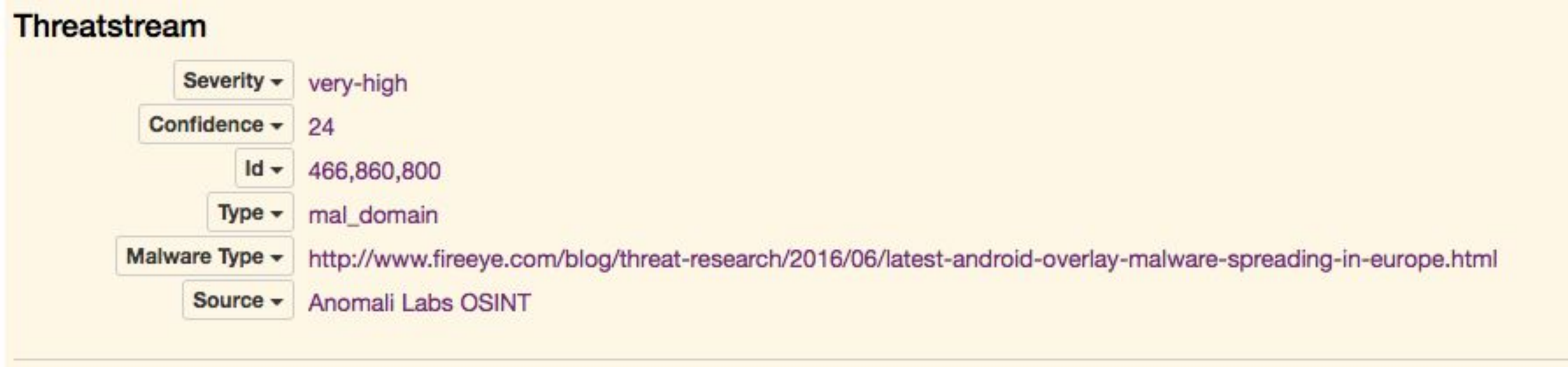
With Intelligence See Everything

Andy Wick

What is it

- Moloch SPI data enhancer
    - Can match on host/domains, md5, url, ip, ja3, email
    - New for 1.5, can now match on almost any field
    - Can set almost any field in SPI data
    - Can add menu options (called right clicks still)
- Supported data sources
    - Simple Files
    - Commercial Services: OpenDNS, Emerging Threats Pro, Threatstream, …
    - Elasticsearch/Redis
    - New for 1.6, Splunk
- Multilayer caching
    - Capture
    - Redis

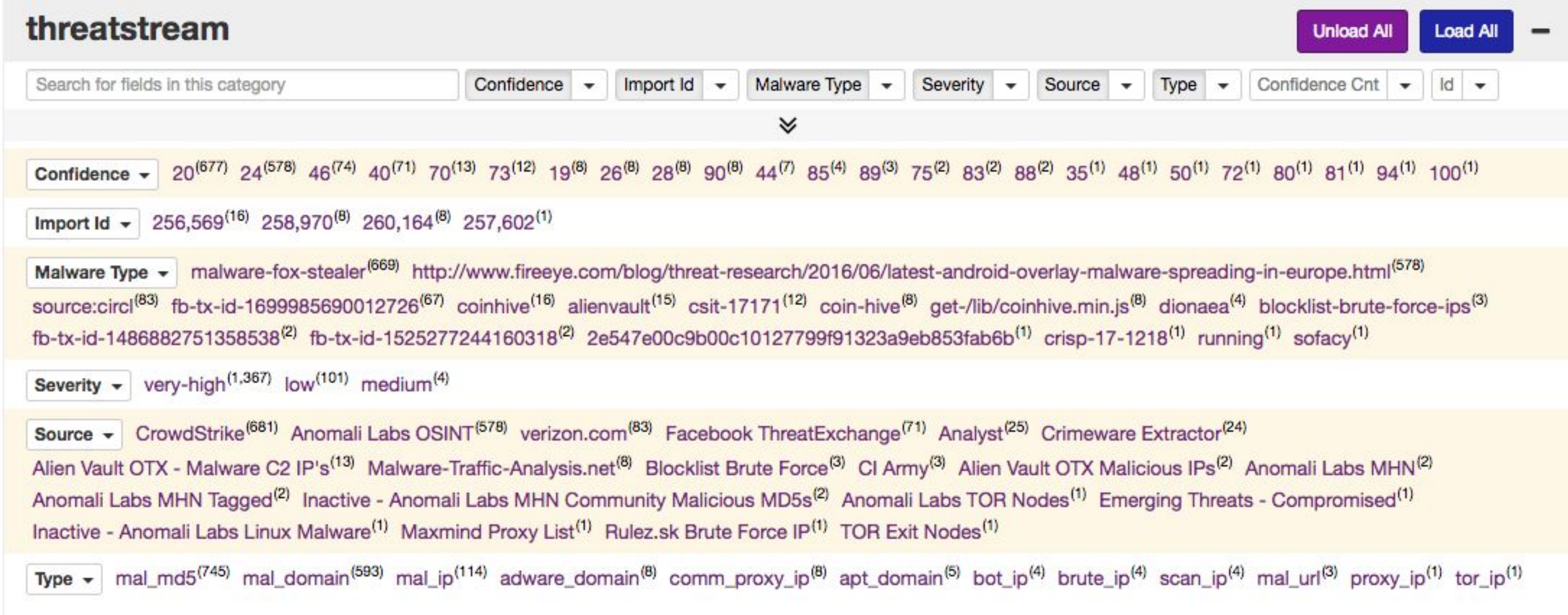
SPI Data Sample - Threatstream



Threatstream

Severity ▾	very-high
Confidence ▾	24
Id ▾	466,860,800
Type ▾	mal_domain
Malware Type ▾	http://www.fireeye.com/blog/threat-research/2016/06/latest-android-overlay-malware-spreading-in-europe.html
Source ▾	Anomali Labs OSINT

SPI View Sample - Threatstream



Confidence 20(677) 24(578) 46(74) 40(71) 70(13) 73(12) 19(8) 26(8) 28(8) 90(8) 44(7) 85(4) 89(3) 75(2) 83(2) 88(2) 35(1) 48(1) 50(1) 72(1) 80(1) 81(1) 94(1) 100(1)

Import Id 256,569(16) 258,970(8) 260,164(8) 257,602(1)

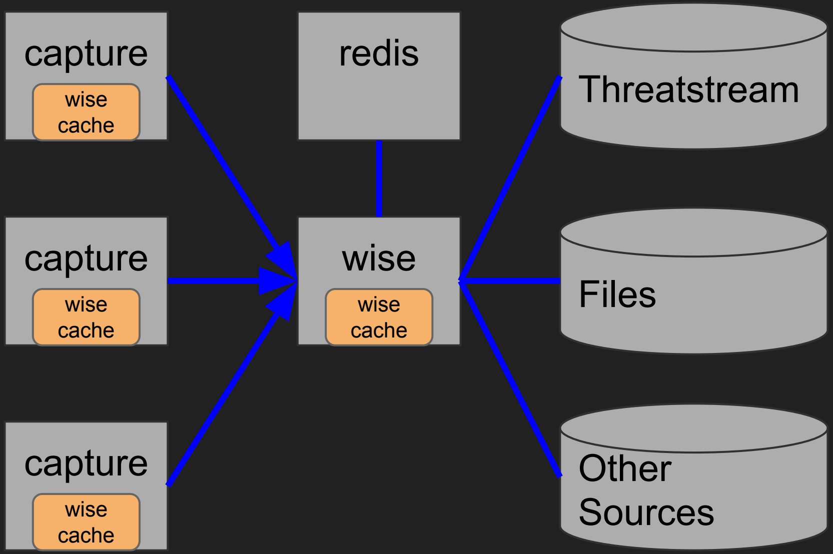
Malware Type malware-fox-stealer(669) http://www.fireeye.com/blog/threat-research/2016/06/latest-android-overlay-malware-spreading-in-europe.html(578) source:circl(83) fb-tx-id-1699985690012726(67) coinhive(16) alienvault(15) csit-17171(12) coin-hive(8) get-/lib/coinhive.min.js(8) dionaea(4) blocklist-brute-force-ips(3) fb-tx-id-1486882751358538(2) fb-tx-id-1525277244160318(2) 2e547e00c9b00c10127799f91323a9eb853fab6b(1) crisp-17-1218(1) running(1) sofacy(1)

Severity very-high(1,367) low(101) medium(4)

Source CrowdStrike(681) Anomali Labs OSINT(578) verizon.com(83) Facebook ThreatExchange(71) Analyst(25) Crimeware Extractor(24) Alien Vault OTX - Malware C2 IP's(13) Malware-Traffic-Analysis.net(8) Blocklist Brute Force(3) CI Army(3) Alien Vault OTX Malicious IPs(2) Anomali Labs MHN(2) Anomali Labs MHN Tagged(2) Inactive - Anomali Labs MHN Community Malicious MD5s(2) Anomali Labs TOR Nodes(1) Emerging Threats - Compromised(1) Inactive - Anomali Labs Linux Malware(1) Maxmind Proxy List(1) Rulez.sk Brute Force IP(1) TOR Exit Nodes(1)

Type mal_md5(745) mal_domain(593) mal_ip(114) adware_domain(8) comm_proxy_ip(8) apt_domain(5) bot_ip(4) brute_ip(4) scan_ip(4) mal_url(3) proxy_ip(1) tor_ip(1)

Architecture



For performance reasons lookups are cached at multiple layers.

1) Check wise cache in capture (ALWAYS)
2) Check wiseService cache (for some sources)
3) Check redis cache (if configured)
4) Ask the data source for information

Capture & Viewer Configuration

```
# Set in [default] and/or for each capture node
wiseHost=wisehost.example.com

# Semicolon ';' separated list of viewer plugins to load and the order to load in
viewerPlugins=wise.js

# Semicolon ';' separated list of plugins to load and the order to load in
plugins=wise.so
```

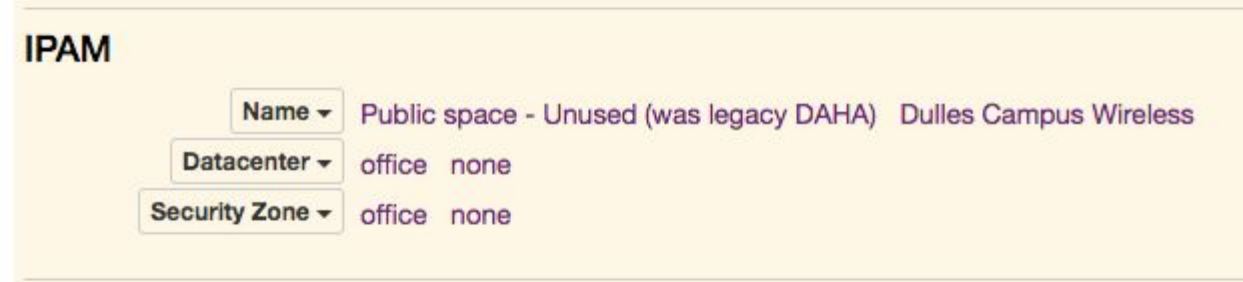
Data source configuration

- Like capture/viewer, everything in an ini file
- Each data source has its own section
  - Some sections are unique like [threatstream]
  - Some sections have prefixes like [file:filename] and [url:urlname]
- Most feeds just require simple configuration with defaults being good enough
- All WISE sources implement some common options
  - cacheAgeMin - For those that cache
  - excludeDomains, excludeEmails, excludeURLs - don’t lookup matching items
  - excludeEmails, excludeURLs - support wildcards
  - excludeIPs - CIDR
- See WISE wiki entry for configuration options

Sample WISE Configuration

```
# wiseService contains global settings and global excludes
[wiseService]
excludeDomains=*.zen.spamhaus.org;*.in-addr.arpa;*.dnsbl.sorbs.net;*.ip6.arpa

[reversedns]
ips=10.0.0.0/8
field=asset

[file:badbadbad.ip]
file=/data/moloch/wisefiles/badbadbad.ip
tags=badbadbad
type=ip
format=tagger
```

IPAM Example

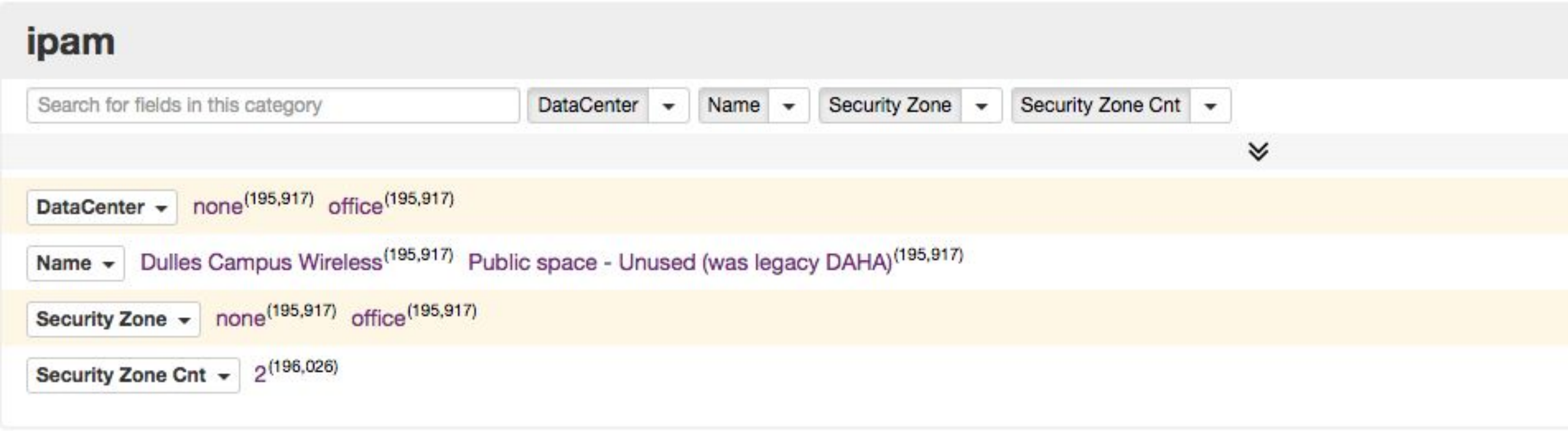


IPAM

Name ▾ Public space - Unused (was legacy DAHA) Dulles Campus Wireless

Datacenter ▾ office none

Security Zone ▾ office none



ipam

Search for fields in this category

DataCenter ▾ none$^{(195,917)}$ office$^{(195,917)}$

Name ▾ Dulles Campus Wireless$^{(195,917)}$ Public space - Unused (was legacy DAHA)$^{(195,917)}$

Security Zone ▾ none$^{(195,917)}$ office$^{(195,917)}$

Security Zone Cnt ▾ 2$^{(196,026)}$

JSON Format - IPAM

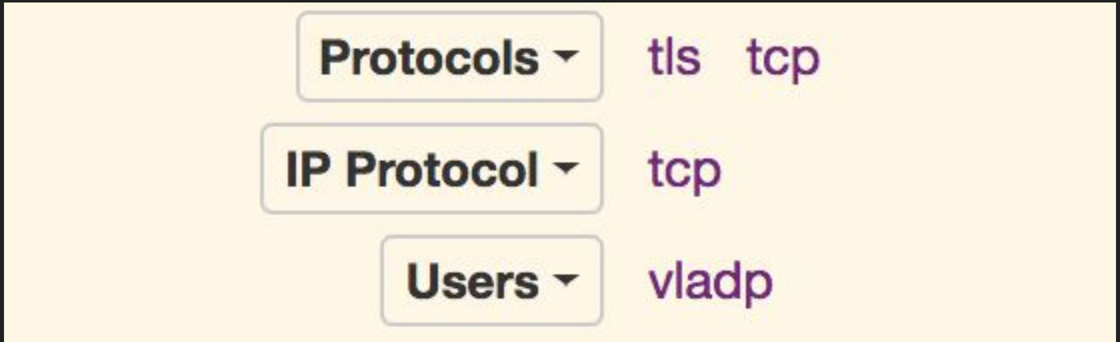
```
[url:ipam]
type = ip
format = json
url = https://exampl.com/getipam.json
reload = 60
keyColumn = CIDR
fields=field:ipam.datacenter;kind:termfield;count:false;friendly:DataCenter;db:ipam.dc-term;help:DataCenter;shortcut:DataCenter\nfield:ipam.zone;kind:termfield;count:true;friendly:Security Zone;db:ipam.zone-term;help:Security Zone;shortcut:SecurityZone
```

JSON Sample Data

```
[{"DataCenter": "none",
"SecurityZone": "none",
"CIDR": "10.0.0.0/8"},

{"DataCenter": "none",
"SecurityZone": "office",
"CIDR": "10.66.0.0/16"}]
```

Tagger Format - badbadbad.ip

```
#field:whatever.str;kind:lotermfield;count:true;friendly:A String;db:whatever.str-term;help:Help for String;shortcut:0

#field:tags;shortcut:1

10.0.0.1;0=this is really bad;1=reallyBadTag
10.0.0.2;tags=anotherRealBadTag
10.0.0.3
```



Protocols ▾ tls tcp

IP Protocol ▾ tcp

Users ▾ vladp

Elasticsearch Source - Get username from panos

```
[elasticsearch:user]
type=ip
onlyIPs=10.10.0.0/16                              = Our VPN space
elasticsearch=https://elk.example.com:9200
esIndex=panos-*                                   = index to search against
esTimestampField=@timestamp                       = what field has the timestamps
esQueryField=sourceIP                             = field to check against
esMaxTimeMS=86400000                              = range of data to search around
esResultField=sourceUserName                      = what json field must exist in results
fields=field:user;shortcut:sourceUserName         = what SPI data fields to set
```

```
{"sourceIP" : "10.10.10.10",
"sourceUserName" : "andywick",
"@timestamp" : "2014-11-13T00:13:32.000Z", ...}
```

Splunk - Table Query

```
type = ip
format = json
host = splunk.host.example.com
port=5500
username={{wise.splunk.user}}
password={{wise.splunk.password}}
periodic=60
query=search index="vpnlog" sourcetype="vpn" assigned earliest=-24h | rex "User <(?<user>[^>]+)>.*IPv4 Address <(?<vpn_ip>[^>]+)>" | dedup vpn_ip | table user, vpn_ip
keyColumn=vpn_ip
fields=field:user;shortcut:user
```

Right clicks

```
[right-click]
VTIP=url:https://www.virustotal.com/en/ip-address/%TEXT%/information/;name:Virus Total IP;category:ip
VTHOST=url:https://www.virustotal.com/en/domain/%HOST%/information/;name:Virus Total Host;category:host
VTURL=url:https://www.virustotal.com/latest-scan/%URL%;name:Virus Total URL;category:url
PTHOST=url:https://passivetotal.org/search/%TEXT%;name:Passivetotal Host;category:host
PTIP=url:https://passivetotal.org/search/%TEXT%;name:Passivetotal IP;category:ip
PTEMAIL=url:https://passivetotal.org/search/%TEXT%;name:Passivetotal User;category:user
```

Creating Views

Instead of

```
this.api.addView("threatstream",
   "if (session.threatstream)\n" +
   "  div.sessionDetailMeta.bold Threatstream\n" +
   "  dl.sessionDetailMeta\n" +
   "    +arrayList(session.threatstream, 'severity-term', 'Severity', 'threatstream.severity')\n" +
   "    +arrayList(session.threatstream, 'confidence', 'Confidence', 'threatstream.confidence')\n" +
   "    +arrayList(session.threatstream, 'id', 'Id', 'threatstream.id')\n" +
   "    +arrayList(session.threatstream, 'importId', 'Import Id', 'threatstream.importId')\n" +
   "    +arrayList(session.threatstream, 'type-term', 'Type', 'threatstream.type')\n" +
   "    +arrayList(session.threatstream, 'maltype-term', 'Malware Type', 'threatstream.maltype')\n" +
   "    +arrayList(session.threatstream, 'source-term', 'Source', 'threatstream.source')\n” )
```

Can now just have one line

```
"require:threatstream;title:Threatstream;fields:threatstream.severity,threatstream.confidence,threatstream.id,threatstream.importId,threatstream.type,threatstream.maltype,threatstream.source"
```

Wise Types

Can now add fields to already created wise types, or create new wise types

This examples add a new “mac” type and adds to the md5 type a new field “blahblah.md5”

```
[wise-types]
mac=db:srcMac;mac.dst
md5=db:http.md5;db:email.md5;db:blahblah.md5
```

Todo

- Make creating new sources easier
- Add UI to see wise state and configuration
- Support multiple WISE servers on one machine better
- Move the source to the top level
- Bro support

QUESTIONS?