RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Share.
Learn.
Secure.
Capitalizing on
Collective Intelligence

Dueling Banjos - Cloud vs. Enterprise Security: Using Automation and (Sec)DevOps NOW

SESSION ID: CSV-W02

Rich Mogull
Analyst and CEO
Securosis
@rmogull
rmogull@securosis.com

Hoff
VP Strategy
Juniper Networks
@Beaker
choff@juniper.net

What We Are Going To Discuss

- Virtualization
- Cloud Computing
- Open Source
- Security Toolsets and Tooling
- Security APIs and Programmatic Operations
- Automation
- Software Defined Security, Compliance & Incident Response
- [Sec]DevOps

What We Are NOT Going To Discuss [In Detail]

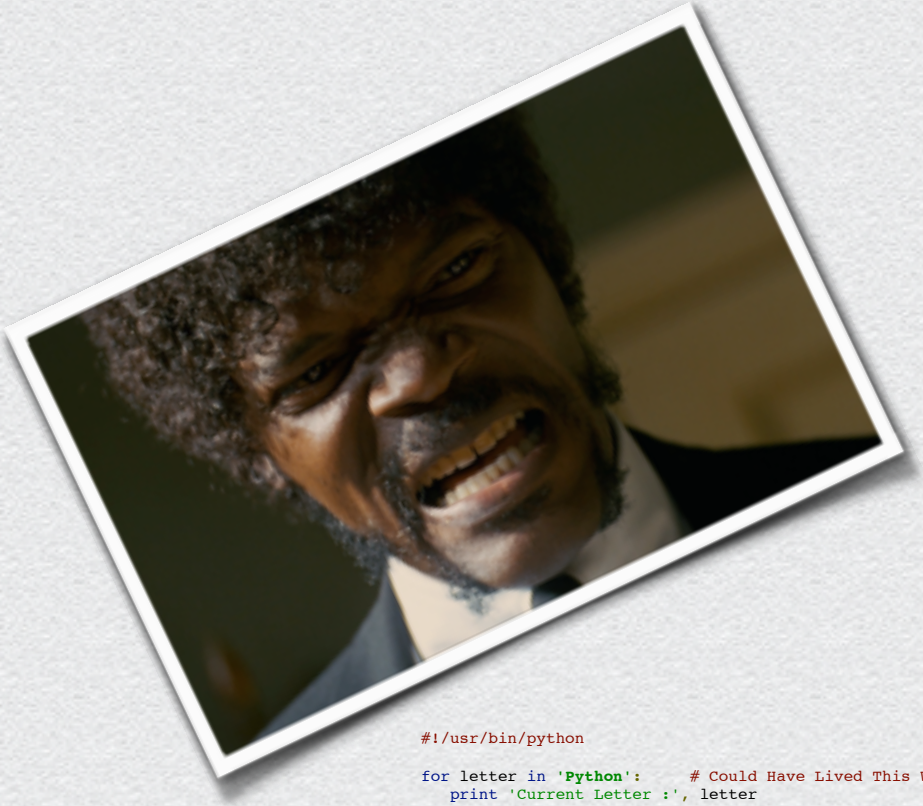
- Developer-centric Software Development Life Cycle
- Vulnerability Assessment
- Code Analysis
- QA/Regression/Unit testing
- Application Language-specific Security
- Implementation specifics of continuous integration/continuous delivery
- …except…

Security Says: “ENGLISH…Do You Speak It!?”

- DART, Ceylon, GO, F#, OPA, Fantom, Zimbu, X10, Haxe, Chapel
- Django, Pylons, Mojolicious
- CouchDB, Hadoop, Neo4J, MongoDB, Cassandra
- Python
- Ruby
- node.js
- Erlang
- Scala
- Clojure
- Groovy

The path of the righteous security man is beset on all sides by the inequities of the selfish and the tyranny of evil developers. Blessed is he who, in the name of scalability and good will, shepherds the weak through the valley of downtime darkness, for he is truly his brother's keeper and the finder of lost vulnerabilities. And I will strike down upon thee with great pwning vengeance and furious anger those who would attempt to poison and destroy my perimeter. And you will know My name is the Compliance Lord when I lay My stateful packet filtering vengeance upon thee.

Developers Say: “CODE…Do You Write It!?”



Say 'Python' again.

Say 'Python' again, I dare you, I double dare you...say ‘Python’ one more time!

```
#!/usr/bin/python

for letter in 'Python':     # Could Have Lived This Way
   print 'Current Letter :', letter
```

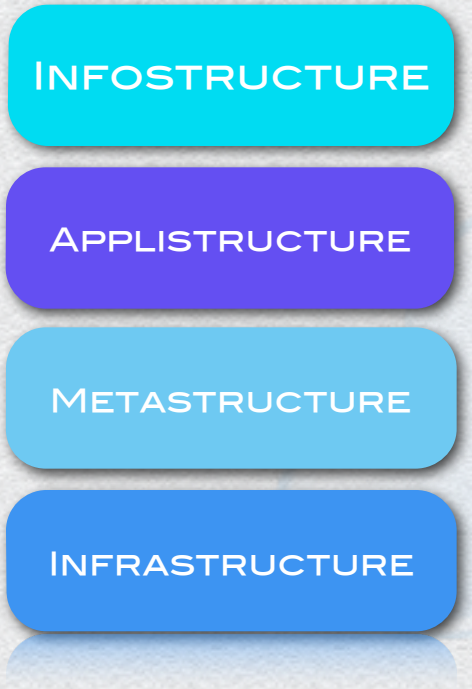
Framing the Problem

- The discipline that is most resistant to change and least likely to adapt is “Security”
- This resistance is usually excused due to a lack of trust and a reliance on people because we don’t trust security automation.
- “Security” continues to rely on a manual supply chain operated by the “Meat Cloud”
- Trustable automation and an operational model to support it is needed

The “Enterprise” vs the “Cloud” Models

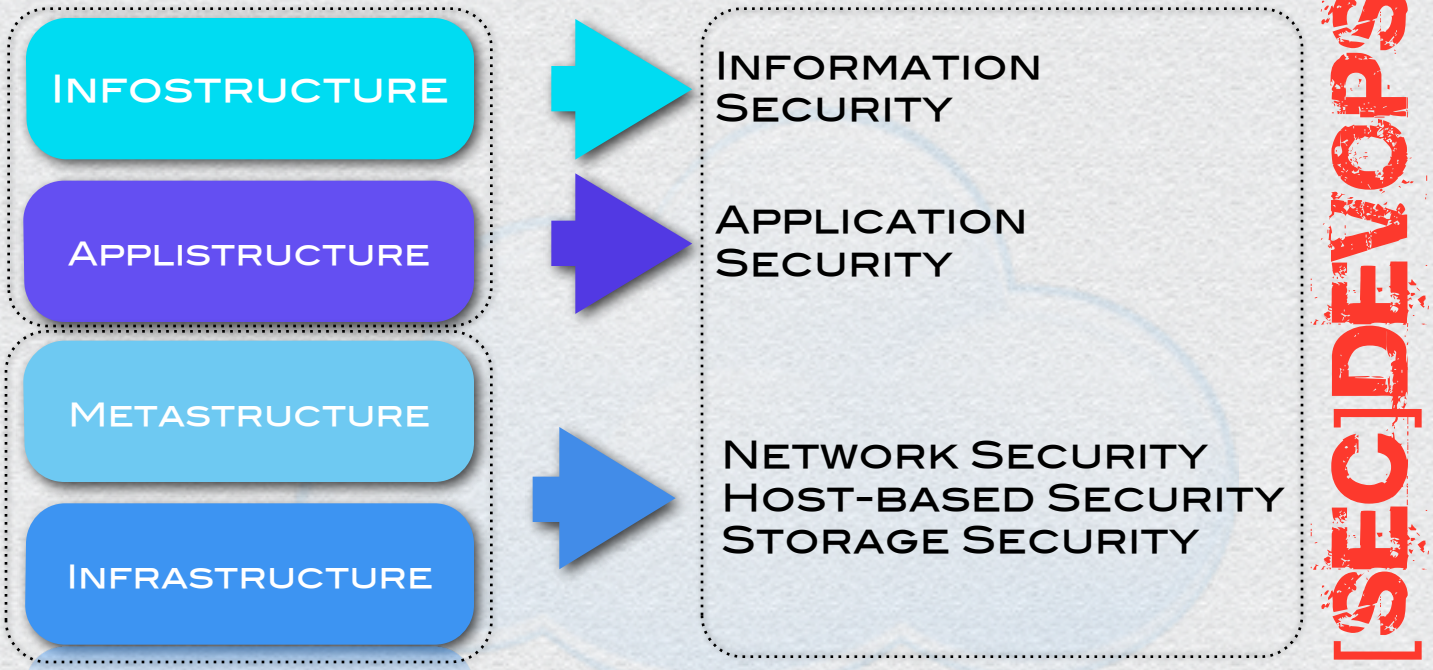
- Cloud is an **operational model**
- DevOps represents an **operational framework**
- Both enjoy their own definitional perversion
- Enterprises are adopting Cloud in various forms; Public/Private/Hybrid, IaaS/PaaS/SaaS
- The traditional silos and organizational dynamics of enterprises — driven by arbitrary economic models — are having a rough time with “DevOps”
- Why? Because **people are conflating the differences in the operational models with the need to adapt their frameworks for servicing it**

IT Deconstructed



- Content & Context - Data & Information
- Apps & Widgets - Applications & Services
- Glue & Guts - IPAM, IAM, BGP, DNS, SSL, PKI & Abstraction layers
- Sprockets & Moving Parts - Compute, Network, Storage

What This Means To Security



The Challenge In Semantics…

- If we don't have consistency in standards/formats for workloads & stack insertion, we're not going to have consistency in security
- Inconsistent policies and network topologies make security service, topology & device-specific
- Fundamentally, we need reusable and programmatic security design patterns; Controls today are CLI/GUI based
- Few are API-driven or feature capabilities for orchestration, provisioning as the workloads they protect

…We ought to think about security like this:

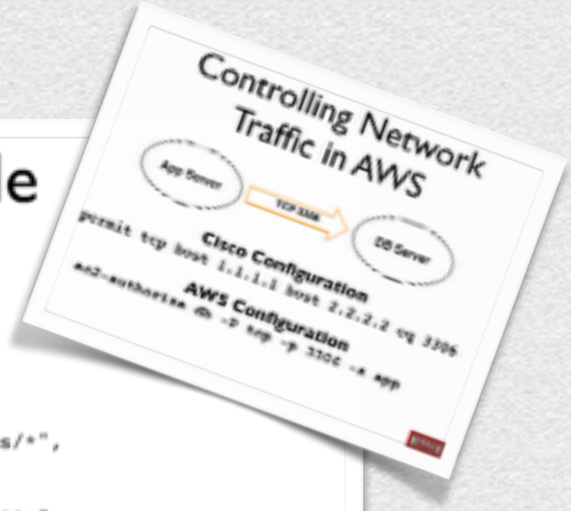
Working with VMware vShield REST API in perl. Richard Park, Sourcefire

…or this…

Policies - Example

```
{
  "Statement": [
    {
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::testbucket/files/*",
      "Condition": {
        "DateLessThanEquals": {
          "aws:CurrentTime": "2012-05-31T12:00:00Z"
        },
        "IpAddress": {
          "aws:SourceIp": "1.1.1.1"
        }
      },
      "Principal": {
        "AWS": [
          "123456789012"
        ]
      }
    }
  ]
}
```

NETFLIX



AWS Security : A Practitioner's Perspective. Jason Chan, Netflix

What's Missing?

- Instrumentation that is inclusive of security
- Intelligence and context shared between infrastructure and applistructure layers
- Maturity of "automation mechanics" and frameworks
- Standard interfaces, precise syntactical representation of elemental security constructs < We need the "EC2 API" of Security
- An operational security methodology that ensures a common understanding of outcomes & "DevOps" culture in general

So?

- Regardless of whether you're an Enterprise or a Cloudyrprise or a Hybridprise, there are various levels of sophistication and maturity that exist
- There are plenty of Enterprises who have their operational security house in order and plenty of Cloudyprises who fall over constantly and vicey-versey
- The Operational Model doesn't dictate the success of the Operational Framework but the converse is true
- Changing how, where and when security is done requires a different framework for doing it. And who does it.
- This is **[Sec]DevOps.**

[Sec]DevOps & The $64,000 Security Question

- What would you do differently — and how — if you took your most important assets from behind your firewall and processes and plugged them directly into the Internet?

- What if these assets are sprinkled around in your virtualized Data Centers, multiple Public Cloud IaaS providers, and linked to one or more SaaS providers — and you need to manage workloads and security…at scale.

- Are you still going to use the Meat Cloud?

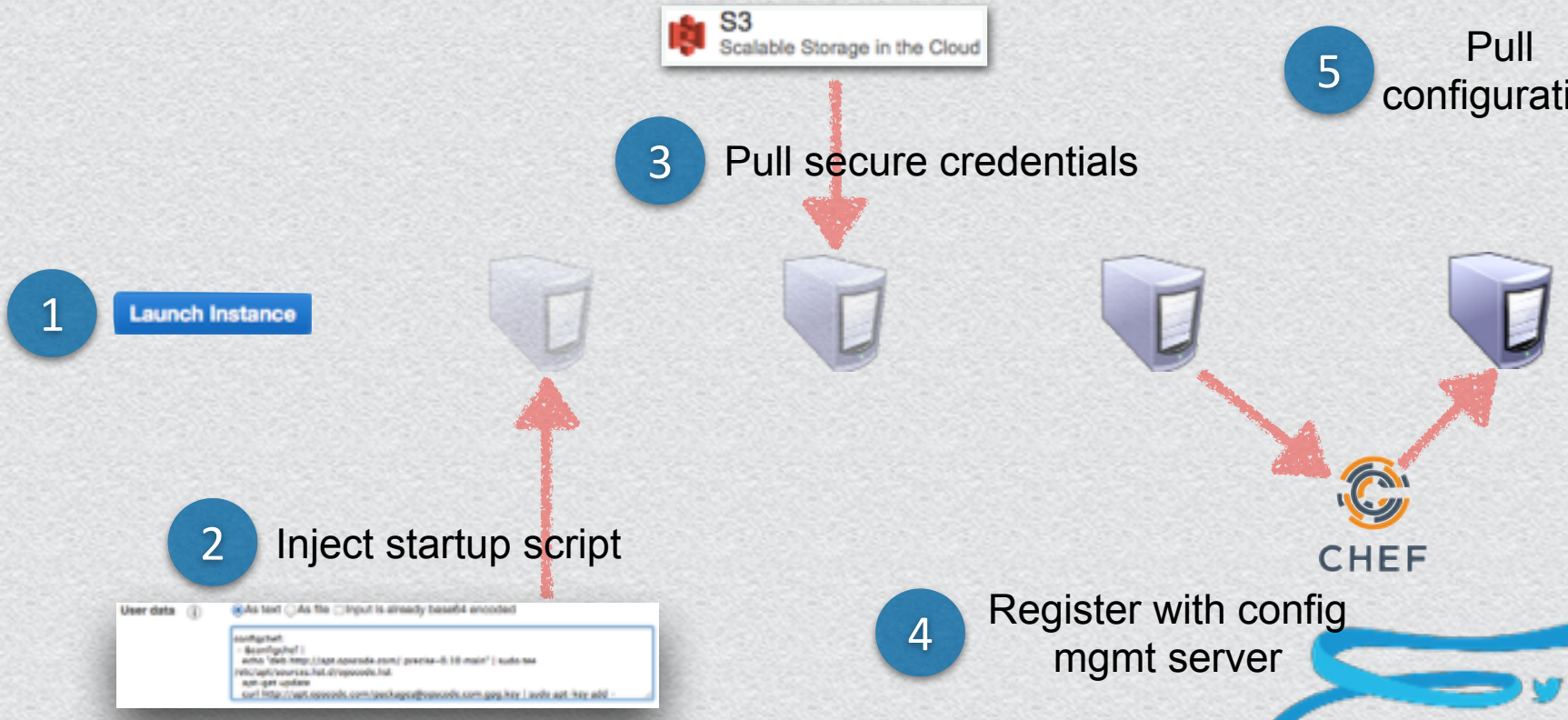
A Real Scenario

The scenario:

- 24 “data centers,” 4 of them connected via a VPN a public IaaS cloud (Hybrid)
- Massive private WAN with complex routing, DNS and load-balanced infrastructure and virtualized overlay networking (SDN)
- 1000 distributed firewalls — a combination of physical & virtual
- 10,000 hosts — bare metal and virtualized with 2 hypervisors
- Custom-written orchestration system
- Internally-deployed, self-service “Private Cloud” and integrated Platform-as-a-Service
- 500 firewall policy changes a week

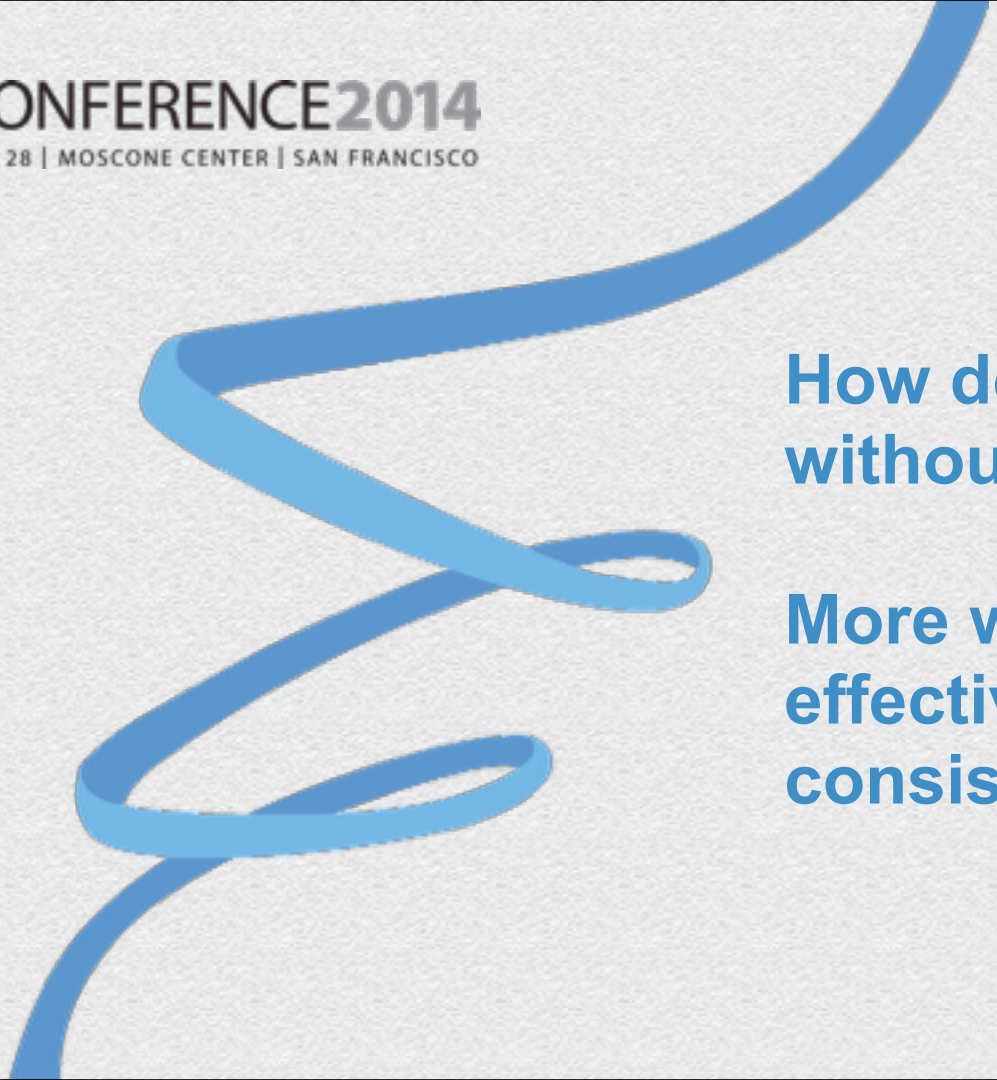
This is real today. We call it Software Defined Security

Welcome to SecOps/SecDevOps



DEMO

Completely automated and consistently and persistently enforced.



How do you do this without automation?

More work, less effective, less consistent.

Software Defined Security in Action

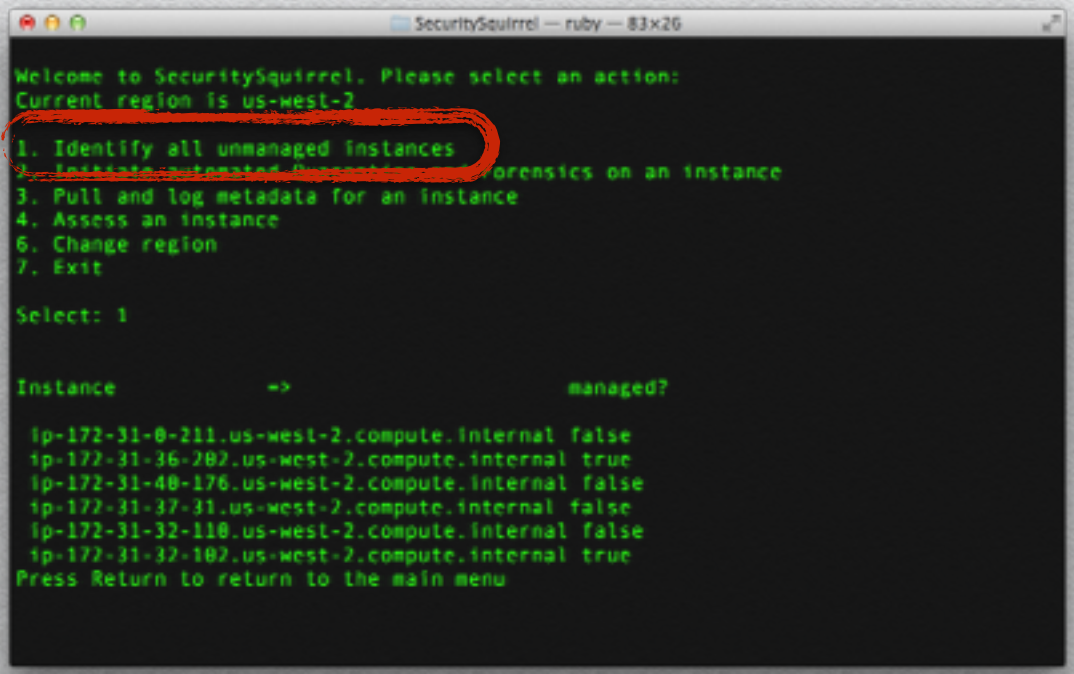
- Meet SecuritySquirrel, the first warrior in the Rodent Army (apologies to Netflix).
- The following tools are written by a short, red-headed analyst with a shorter temper and a Ruby-for-Dummies book.
- Automated security workflows spanning products and services.

Problem: Identify Unmanaged Servers

1. Scan the network
2. Scan again and again for all the parts you missed
3. Identify all the servers as best you can
4. Pull a config mgmt report
5. Manually compare results

The Software Defined Security Way

DEMO



1. Get list of all servers from cloud controller (can filter on tags/OS/etc).
   - Single API call
2. Get list of all servers from Chef
   - Single API call
3. Compare in code

Problem: Compromised Server Incident Response

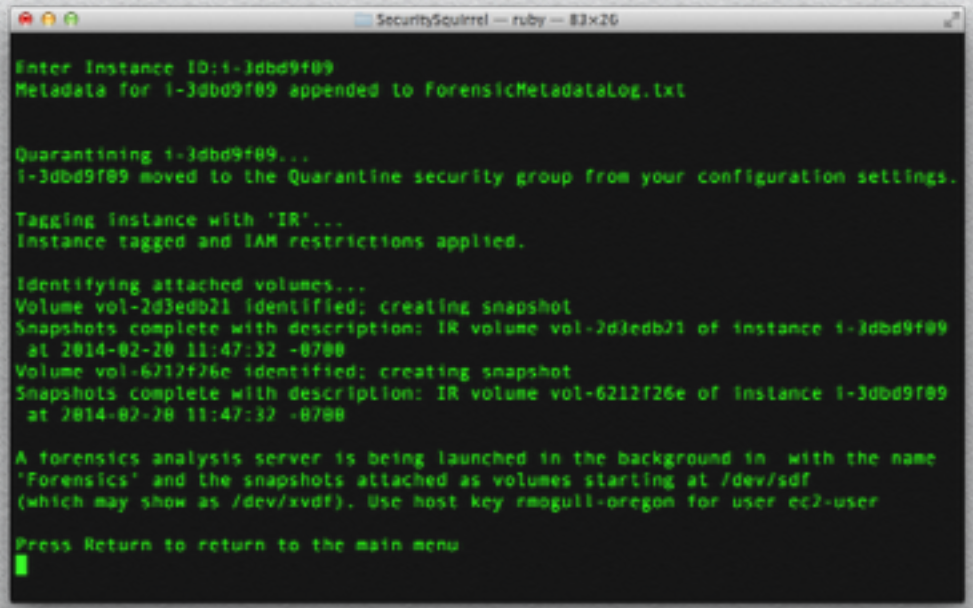
1. Detect Compromise
2. Pull server information (If you have it)
3. Quarantine
4. Image
5. Analyze
6. Recover

= Hours!

Each step is manual, and uses a different set of disconnected tools

The Software Defined Security Way

DEMO



1. Pull metadata
2. Quarantine
3. Swap control to security team
4. Identify and image all storage
5. Launch and configure analysis server
6. Can re-launch clean server instantly

The Only Difference is the APIs and Program Flow

A Software Defined Security Rainbow Unicorn

- Automating a secure vulnerability assessment involving a cloud service and two commercial security products.
- Open firewall, open host firewall, trigger scan, close firewalls.

QUALYS® CONTINUOUS SECURITY

CloudPassage

amazon web services™

Our Call to Action…

I, Network Engineer

s/Network/Security

An Impatient Start

Let me make a couple of things clear right from the outset:

1. ***I am not a programmer!*** Yes I have written the odd script here and there in the deep dark past, but I am by no means a programmer. All of my scripts have been about automating some task I had to do. As long as it worked, I didn't care how efficient or pretty it was – it did what I needed.
2. ***I have no intention of becoming a full time programmer!*** I like being a network architect and I like building and playing with network toys. All I want is the ability to make my job easier, which leads me to my last point...
3. ***I am lazy!*** I don't like repetitive work. I would rather do something once or twice and move on. Computers are here to do the mundane stuff for us, so we can create more awesome. I would rather write scripts for other people to do it next time instead of bugging me about it.

So with the above three stipulations in hand, I started to learn Python. Now, when I say "started", I literally mean a week ago. I already knew the basics of loops and conditionals etc, but I couldn't read a lick of Python this time last week. I tried the various online tutorials such as over at Code Academy and Learn Python the Hardway, but I knew the only way I was going to get my head around Python was to jump in and just start coding the working I had on my plate.

***Kurt Bales**, Senior Network Engineer blogger at "www.network-janitor.net"*

*Borrowed from Jeremy Schulman, Juniper Networks

An Engineer's Approach:

- Get started "day one" using Python interactive shell
- Do it the way a network engineer thinks and interacts with the network, not like a Programmer/API
- Do not require knowledge of XML, Junos, NETCONF
- Give me "CLI access" if I get stuck, but no CLI screen-scraping
- Give me access both config and operational data in standard Python types like dictionary (hash) and list
- Make it Open-Source so I don't have to wait for "The Vendor" to add/ fix things

*Borrowed from Jeremy Schulman, Juniper Networks

If Yan Can Cook, You Can Too!

RSA CONFERENCE 2014
FEBRUARY 24 – 28 | MOSCONE CENTER | SAN FRANCISCO

Share.
Learn.
Secure.
Capitalizing on
Collective Intelligence

Dueling Banjos - Cloud vs. Enterprise Security: Using Automation and (Sec)DevOps NOW

SESSION ID: bla bla

Rich Mogull
Analyst and CEO
Securosis
@rmogull
rmogull@securosis.com

Hoff
VP Strategy
Juniper Networks
@Beaker
choff@juniper.net

http://github.com/securosis/securitysquirrel