RSA®Conference2015

San Francisco | April 20-24 | Moscone Center

CHANGE

Challenge today's security thinking

SESSION ID: CSV-F03

Realities of Private Cloud Security

Scott Carlson – PayPal

@relaxed137

#RSAC

PayPal Cloud & Software Defined Data Center

VIRTUAL

Cloud Design Principals, traditional Data Center
Deploy from Templates
Any Image, Anywhere

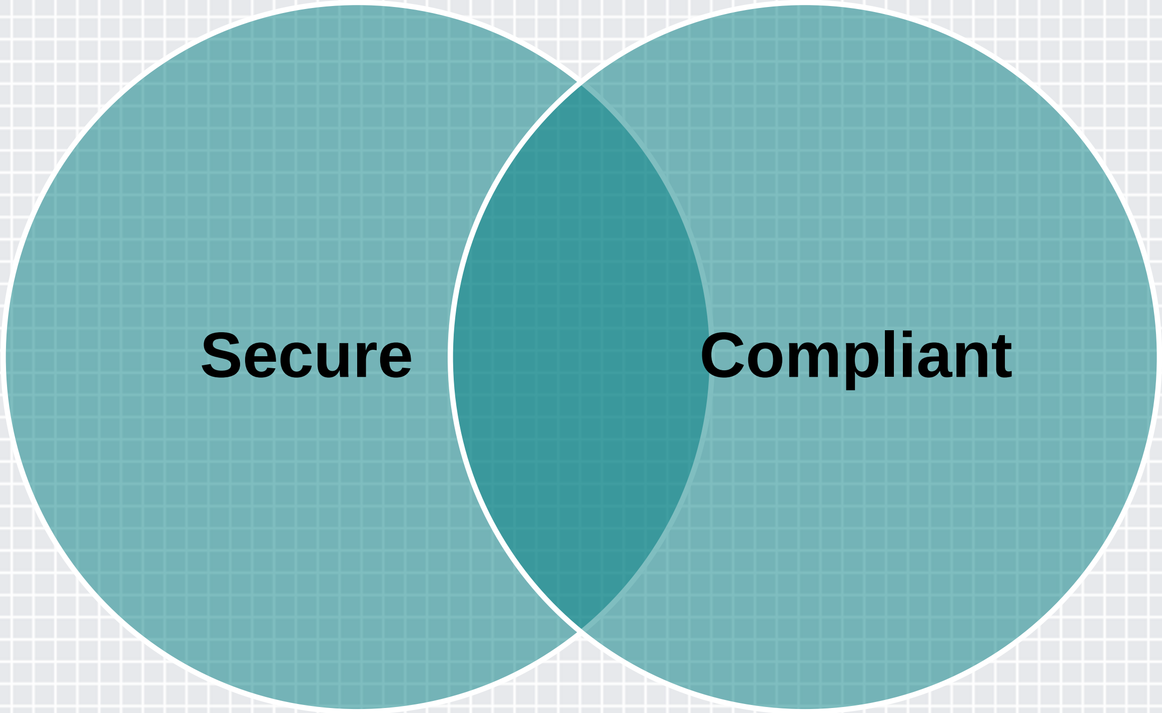
ELASTIC

Automatically scale up/down workloads
Follow devops auto-deployments CI/CD
Respond to intra-cloud events

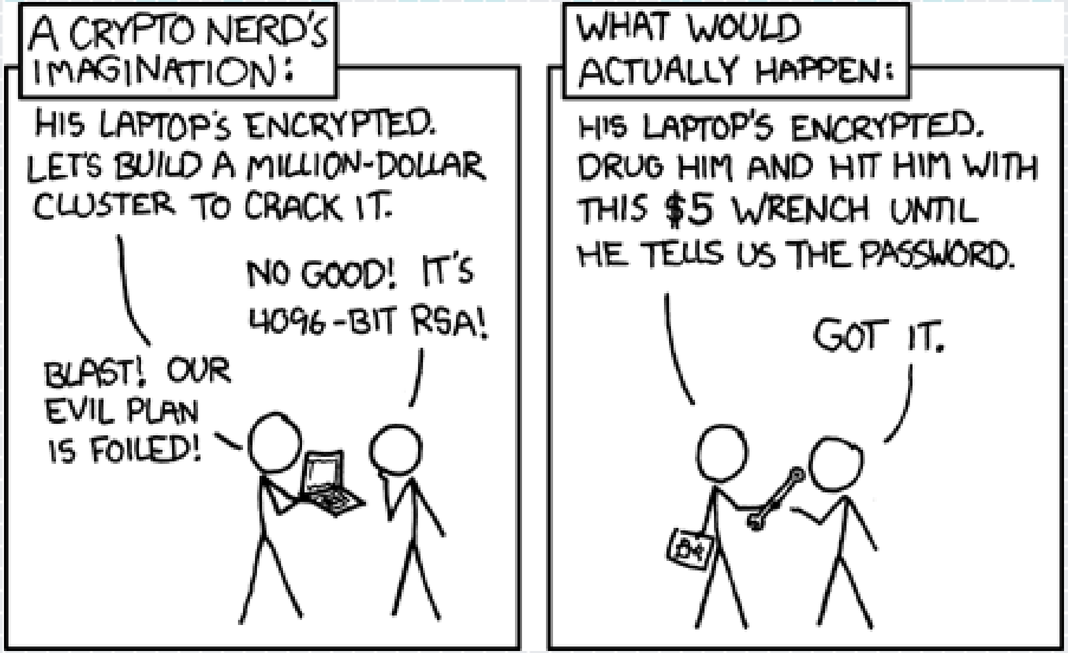
SECURE

PCI-DSS 2.0 and 3.0
Local Country Requirements

Compliance <> Security



PCI DSS
HIPAA / HITECH
SOX
FedRAMP/FISMA
ISO/IEC 27001-2005
NIST SP800-53


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LET'S BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.

@ http://xkcd.com used with permission under Creative commons License

Compliance Requirements of PayPal

- ✓ Compliant with PCI-DSS 2.0 Standards
- ✓ Non-US locations compliant with local country regulations

What is *Secure*

PREVENT

Be patched, be compliant, be hardened, be layered, don’t let data leave your network

DETECT

Log it all; parse it all; sesame street logic; leave no stone unturned

RESPOND

Quarantine; active defense; mitigate; high priority patches; bug fixes; block ports; kill data streams; sever connections

Are there any Private Cloud Standards?

… Hardly …

Cloud Controls Matrix 3.0.1 – release date 7/2015

https://cloudsecurityalliance.org/download/cloud-controls-matrix-v3-0-1/

PCI DSS Virtualization Guidelines

https://www.pcisecuritystandards.org/documents/Virtualization_InfoSupp_v2.pdf

PCI DSS Cloud Computing Guidelines

https://www.pcisecuritystandards.org/pdfs/PCI_DSS_v2_Cloud_Guidelines.pdf

NIST Special Publication 800-144

http://csrc.nist.gov/publications/nistpubs/800-144/SP800-144.pdf

What about Public Cloud?

A lot of vendors have been upping their game with security in the public cloud. If your organization does not have an Information Security organization, you do not run your own data center, or you just flat out don't know where to begin, there are public cloud solutions that can meet most information security requirements today.

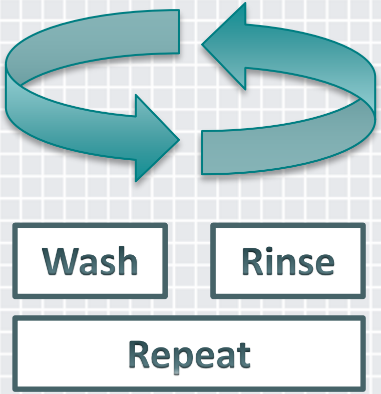
… This talk is not about the Public Cloud …

Our Methodology – Private Cloud Security

- **Don’t make it too hard – it’s just mostly infrastructure**
- **Adapt FAST**
- **Developers & “AAS” Washing changes the access paradigm (2FA + RBAC anyone?)**
- **3rd-party applications sitting on a cloud are still just 3rd-party applications**
- **Home-grown application sitting on a cloud are still just home-grown applications**
- **Cross-managing “in-scope” and “out-of-scope” environments is fraught with peril. Tread Lightly…**
- **The minute something is created in the cloud, it is subject to the requirements. Existence = *production* because it *could* be compromised, attacked, …**

Agile

Quick and well-coordinated in movement; marked by an ability to think quickly; intellectual acuity



- Consider everything dirty… examine it… spray the bad parts… clean it… use machines to do the dirty work
- Run traffic over it… verify assumptions… send it back to the wash if needed… deliver to customer… use it yourself
- Check your work… check new versions… talk to new people… find all of the new and exciting ways people are doing things

Basic Methodology

Just pretend it is infrastructure

OpenStack has servers in it

- Hardware configured and dedicated to the cloud
- Hypervisor/Build Image meeting NIST/CIS standard templates
- Vulnerability Scanning with third-party tooling
- Patching 7-, 30-, 90-day windows with vendor provided patches
- Configuration Management for important system files
- Password Management – non-default, complex and unique!

OpenStack has users in it

- Do not use shared accounts for anything. Just don’t
- Log everything (auth) about a user. Send it somewhere you can find it. Keep it a LONG time
- Define a Role, Add People to the Role, Attest the Role
- Multi-Factor Tokens!!!!
- All cloud actions tied back to an individual user

Basic Methodology

Just pretend it is infrastructure

Hypervisor Components

- It’s just “Linux”. Treat it like hardened “Linux” and lock it down to standards (CIS, NIST)
- Have a separate management interface from your production traffic (physical or virtual)
- Do not combine security zones within a single hypervisor because then it’s ALL “in-scope”
- Audit access, audit changes, be ready to show your work
- Be ready to defend decisions to share ports for components

OpenStack Software Stack

- Limited vulnerability scanning in a programmatic way, have to build your own (Fortify, AppScan)
- Getting code from Trunk = Open Source Happiness, but have your licenses reviewed!
- You still need to code review if CDE passes through here
- Avoid Avoid Avoid actual data getting put in your cloud stack

  (not guest VMs, those are ok)

Basic Methodology

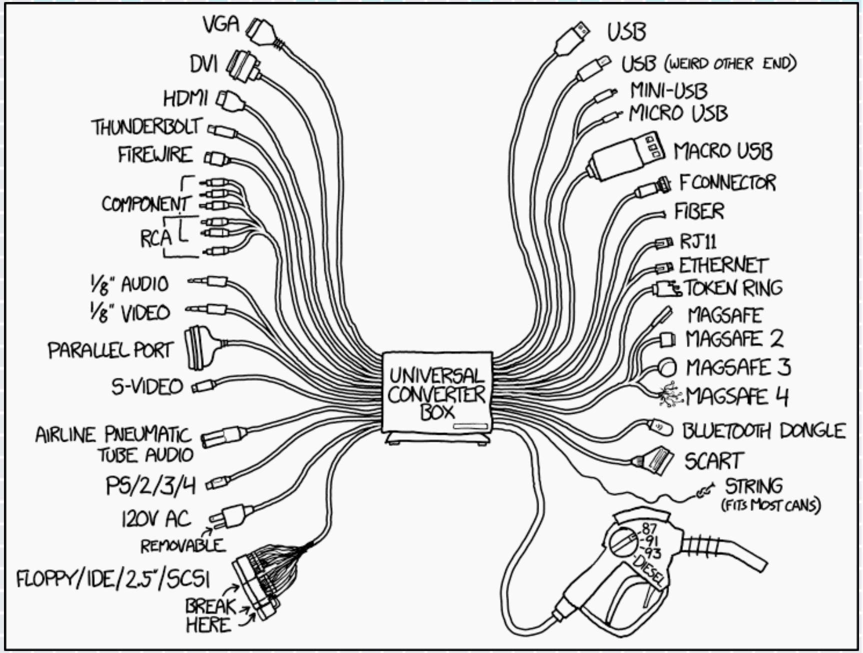
Just pretend it is infrastructure

Physical Network Components

- Firewall rules around the cloud to limit ingress and egress (is iptables a firewall?)
- Monitor what happens on your firewalls, send it somewhere, keep it a LONG time
- Make sure the person building your network isn't the person building your cloud (SOD)
- Configuration guidelines exist for most physical installations (avoid virtual for now…)
- Automation is fine, but make sure you log it and auto-ticket it

Virtual Network Components

- Too early in the testing process to rely on virtual versions of components at scale [No standards, Auditor Support]
- Okay for intra-tenant traffic with minimal rule set
- Same rules for physical apply to virtual. Has your third-party pen-tested and certified their thing?



@ http://xkcd.com used with permission under Creative commons License

Basic Methodology

Just pretend it is infrastructure

Data

- If it’s card-holder data, controls become interesting very quickly
- Encrypt your data PRIOR to writing it to databases/repositories
- Storing things encrypted at rest in VMs mean you can’t use OpenStack components
- HSM, crypto, key management required
- User management, controls over data, logging, all of the standard stuff needed

Basic Methodology

Just pretend it is infrastructure

“As a Service” Tooling

- Role-based access
- 2FA to “in-scope” environments
- TLS on EVERYTHING
- Don’t pass your tokens around, expire them quickly
- Don’t rely on credentials from the wrong environment!
- Always check for software vulnerabilities in your code
- NOTHING goes into production without passing the sanity test… NOTHING

Basic Methodology

Just pretend it is infrastructure

Central Management Considerations

- Make all of your management control planes trusted
- It is NEVER okay to talk from a lower-trust environment into a higher-trust environment
- It is better to expose a protected web front-end than keep the “manager” in an untrusted zone
- If your control plane is compromised, your cloud is toast. Cloud expands the cascade of compromise across your whole environment depending on your control plane and availability zones
- Deleting your cloud as an attack is just the same as a compromise sometimes

Secure is not a permanent state

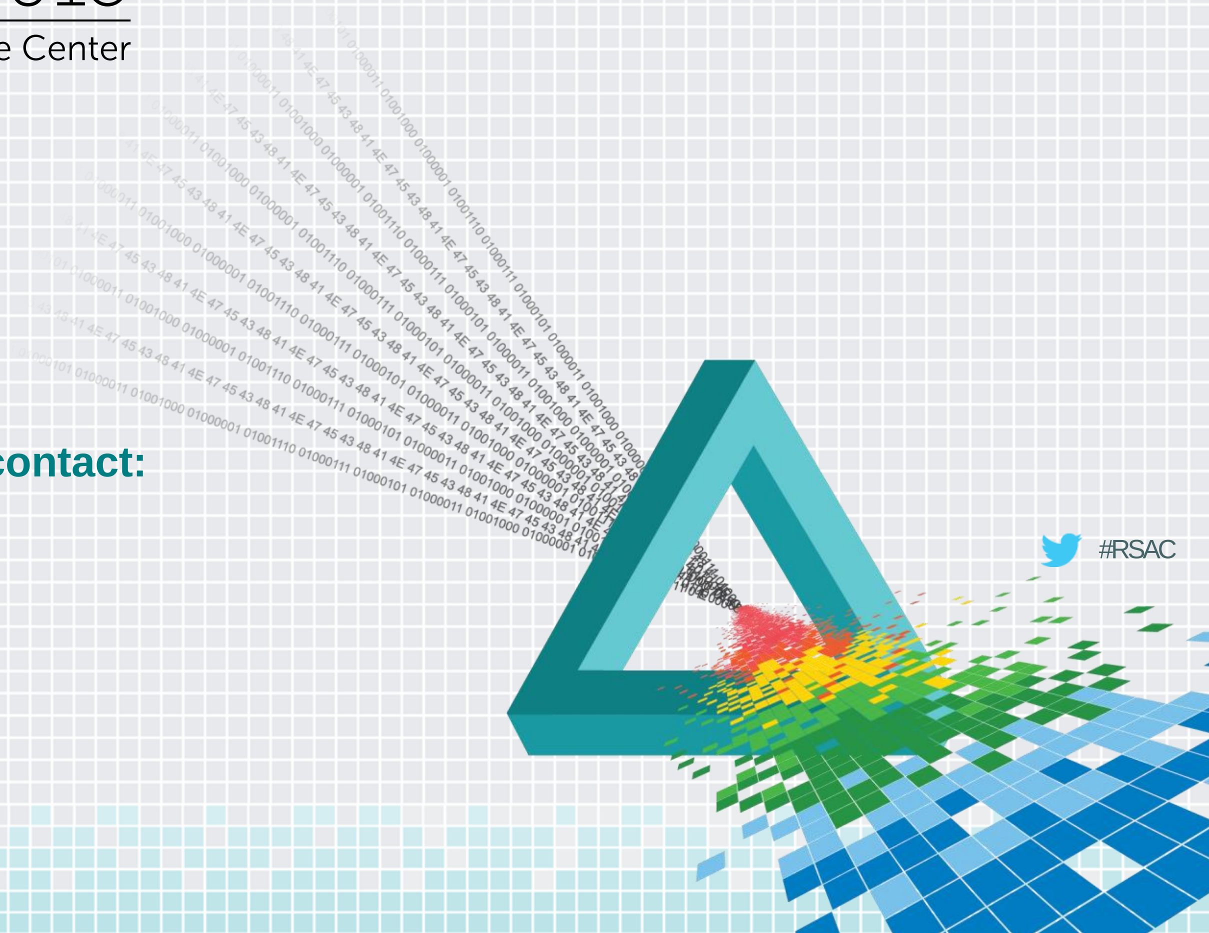
Security can not work effectively unless you have *agility*

How to apply this to your own environment?

- Look into how you are using your cloud today
- Verify the security controls and regulatory controls you need
- Do the basics discussed here
- Pen Test your cloud
- Resolve the findings, and then start iterating and making more secure!
- Get involved to fix industry standards so that private clouds are people too!

RSAConference2015

San Francisco | April 20-24 | Moscone Center



For more information, please contact:

Scott Carlson

sccarlson@paypal.com

@relaxed137

#RSAC