



Securosis
2012

Presents

Cloud-Sec 12 Step

Adrian Lane
Securosis, LLC

Outline

- Cloud Overview
- The '12 Steps'
- Recommendations

Cloud Overview



What 'It' Is

- Abstraction of Infrastructure
- Elastic and Dynamic Resource
- Resource Democratization
- Services Oriented Architecture
- Utility Model of Consumption & Allocation

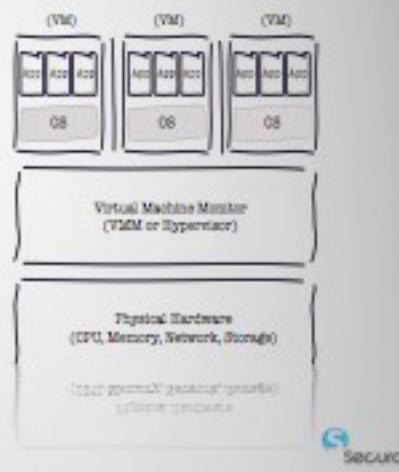
NIST Model of Cloud Computing

Visual Model Of NIST Working Definition Of Cloud Computing
<http://www.csrc.nist.gov/groups/SNS/cloud-computing/index.html>

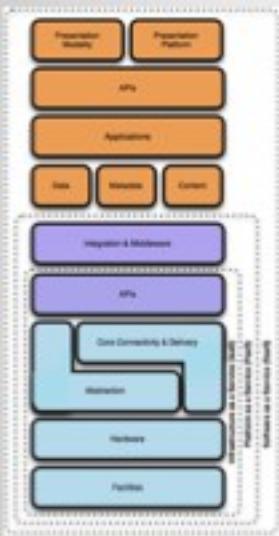


What ‘It’ Is Not

- It's not mainframe, 'cloud in a box' nor virtualization



Cloud Service Models

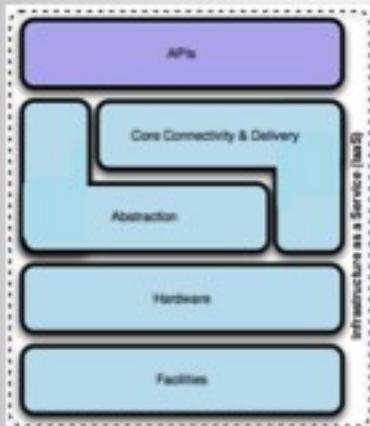


- Software as a Service (SaaS)
- Platform as a Service (PaaS)
- Infrastructure as a Service (IaaS)

*Graphics courtesy Chris Hoff - Rational Survivability

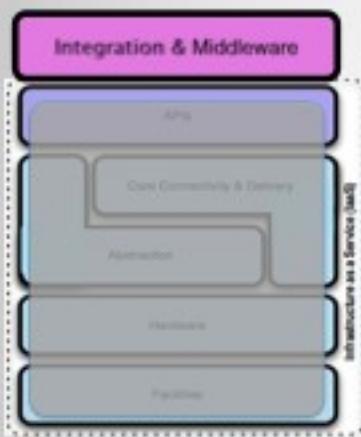


Infrastructure as a Service



- Provider supplied network, computing and storage.
- Customer deploys & configures software
- Amazon EC2, Rackspace, GoGrid

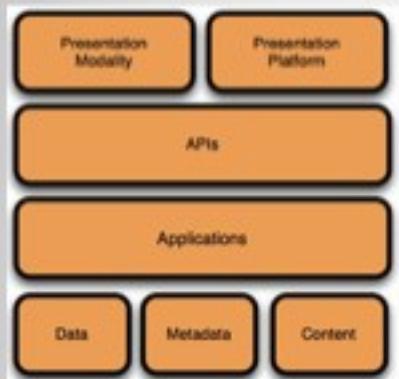
Platform as a Service



- Provider supplied integration & middleware, abstracted resources
- Database, IAM, Messaging & Mgmt APIs
- Force.com, Azure, Database.com, Google AppEngine



Software as a Service



- Provider supplies application and abstracted resources
- Consumer supplies data
- Salesforce.com, Google Apps



Security Implications

- Variable control
- Variable visibility
- Variable resource availability
- Variable simplicity



Which Means ...

Security is different than traditional IT

- Can't test in the same way
- Vulnerabilities to data are different
- Access to logs is limited
- Security models are different
- Limited by what you can deploy



Focus

- PaaS & IaaS (This is OWASP & you're not programming SaaS environments)
- Development oriented view
- EC2: Easy access, mature API, developer friendly., SOA, Cheap. Not an endorsement, it's just easier this way.
- Public or hybrid deployments



A 12 Step Program -

For Cloud Security





Application Security



What We'll Cover



Identity and Access



Code Analysis



Deployment & Configuration



Design



Architecture



Testing



Threat Modeling





Application Security & SDLC



- SOA by design & functional segmentation
- IAM & federation scheme
- Threat/Risk Analysis

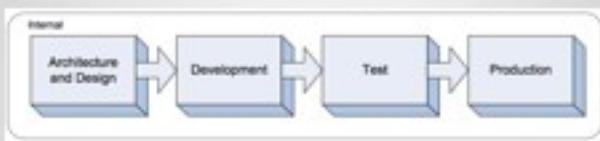


1. Architecture

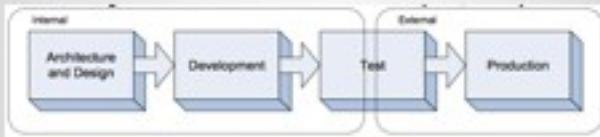


Trust Boundaries

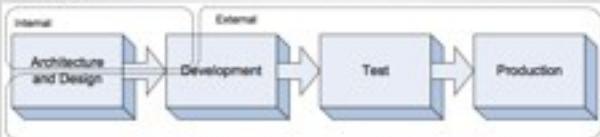
Traditional



IaaS



PaaS



Design for Threats

Example Mapping Threat Model to Countermeasures

Threat	Security Service
Spoofing	Authentication
Tampering	Digital Signature, Hash
Repudiation	Audit Logging
Information Disclosure	Encryption
Denial of Service	Availability
Elevation of privilege	Authorization



App-Dev Lifecycle Security Issues

- App Stack control
 - What you can test
 - Instance validation
 - Vendor API's and abstraction
 - Multi-tenancy
 - Different threat models



App-Dev Lifecycle Security Issues Cont.

- Threat models are different as your zones of control are different
- Geared for SOA (Interfaces, IAM, IPC)
- Auditing
- Elasticity~dynamism
- Your 'provider' and service levels





3. Identity & Authorization



- Registration
- Propagation
- User Management
- De-provisioning
- Audit

Identity





- Access Control Policies
- Authentication
- Authorization
- Audit

Access



- Vendor supplied system?
- In-cloud or hybrid?
- Do you really want to pass credentials?
- SSO? Then plan on federated identity

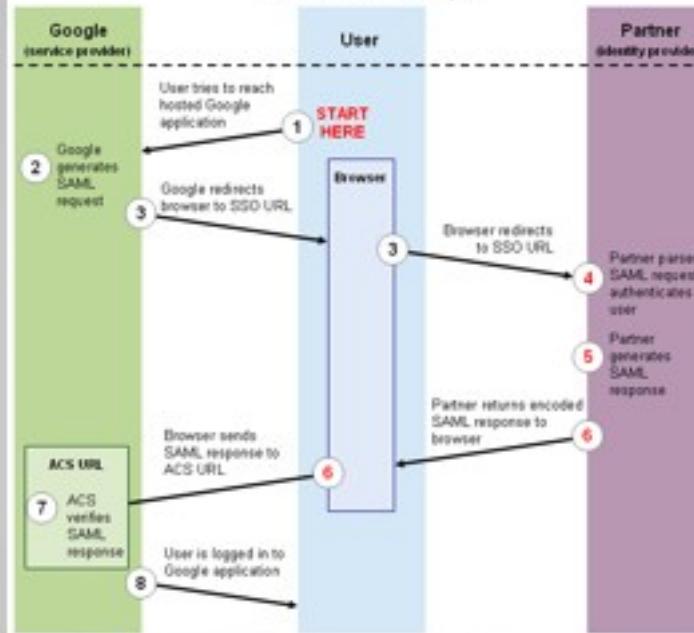


Identity Management Concerns



Figure 1: Logging in to Google Apps using SAML.

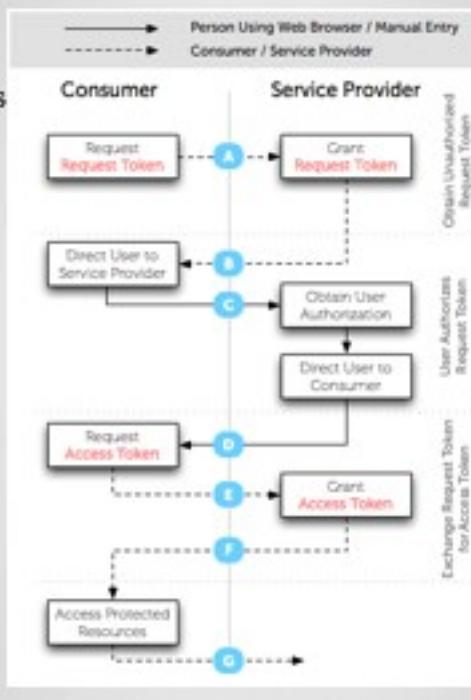
SAML Transaction Steps



SAML Overview



OAuth Process



Options

- SAML - Emerging standard for federated identity
- OAuth - Token based authentication - avoid passing user/password
- WS-Federation - Integrates with WS-* stacks
- Open ID - Webapp only - not really secure



4. Code Analysis

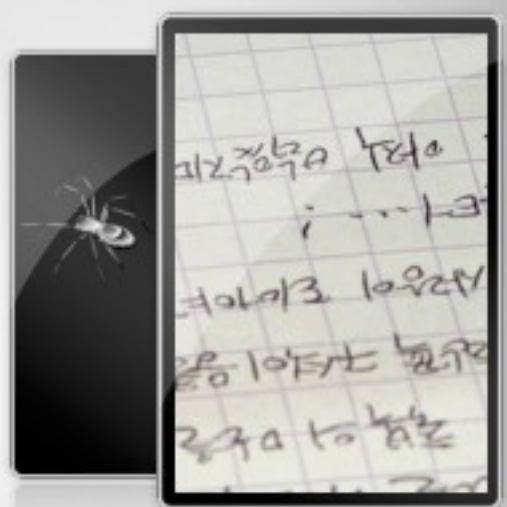
Static Analysis



- IDE Integration
- Prioritization
- Code Coverage
- VA
- DAST Integration
- Platform Neutral



- Leverage private cloud
(Dev vs Prod)
- Fuzzing
- Dynamic Analysis
- Pen Testing
- SLA's



5. Testing



6. App Protection

- WAF - Network config, deployment models, provider
- Honeypots - Stack embedded solutions
- Firewalls - Inherent to provider/deployment



	Developers	Security
1995	CGI, PERL	Network firewalls, SSL
1997	ASP, JSP	Network firewalls, SSL
1998	EJB, J2EE, DCOM	Network firewalls, SSL
1999	SOAP, XML	Network firewalls, SSL
2001	Rmi, RQA	Network firewalls, SSL
2003	Web 2.0	Network firewalls, SSL
2007	Cloud Computing	Network firewalls, SSL

This? Not so much.

Chart courtesy of Gunnar Peterson



Information Security



Data Security Lifecycle



Securoris

- + You're not worried about backup tapes and missing hard drives
- You are worried about snapshots of entire system going public



7. Encryption

Securoris

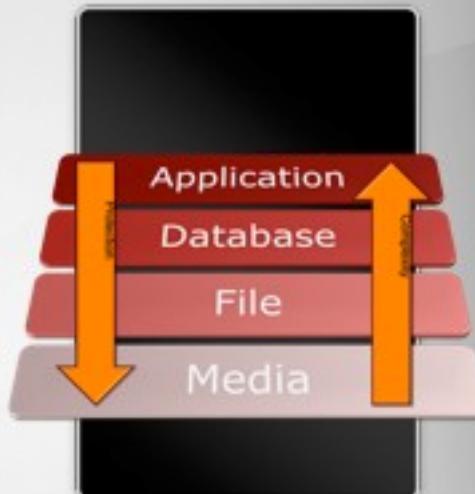
- Moving data into the cloud
- Enforcing rights management
- Securing stored data
- Crypto shredding



Encrypting Application Data



- Application Layer - more work but cloud independent
- TDE - Transparent Database Encryption
- Volume - simple archive security



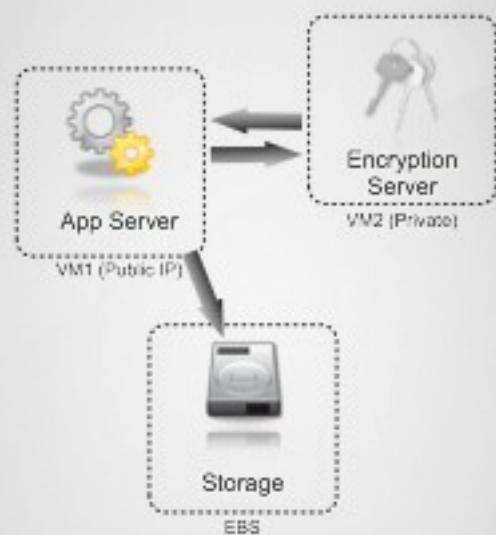
Encryption





- Keys and certificates are basic authentication - not user name and passwords
- Easy to leave key in image
- Local or hybrid key server

8. Key Management



Application Layer Encryption

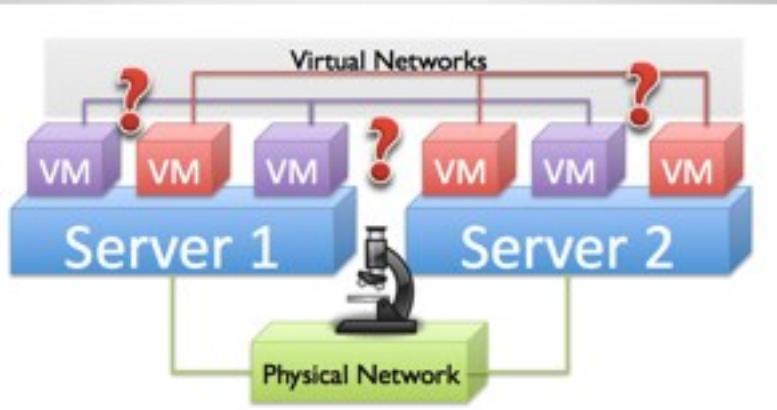




Monitoring

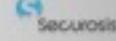


Loss of Network Visibility



Monitoring Technologies

- WAF
 - DAM
 - FAM
 - DLP
 - Service Monitors
- What's provided by vendor?
- What deployment model?



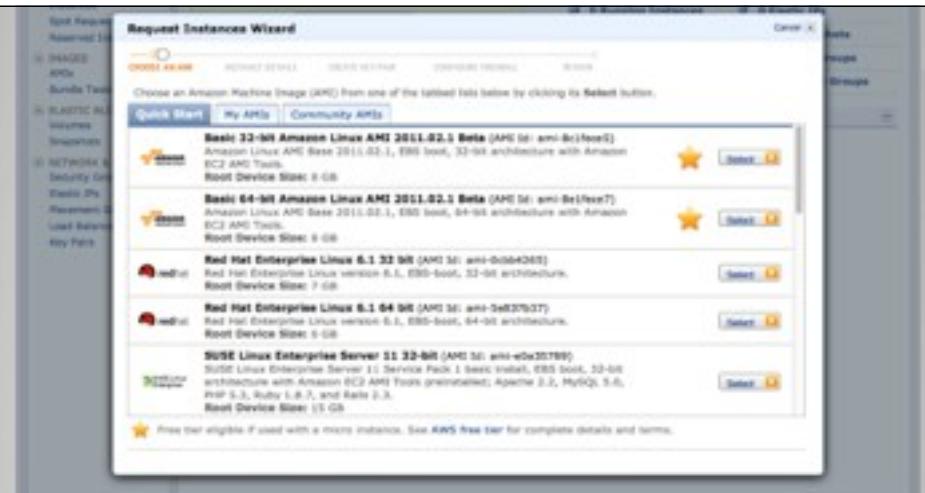
Infrastructure Security



- Trusted Images
- Patching
- Configuration

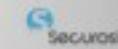


10. Deployment Stack Hardening



Trusted Images

10. Deployment Stack Hardening



This screenshot shows the 'Advanced Instance Options' section of the AWS Lambda configuration interface. It includes fields for Kernel ID, RAM Disk ID, Monitoring, User Data, Termination Protection, and Shutdown Behavior.

Advanced Instance Options

Here you can choose a specific kernel or RAM disk to use with your instances. You can also choose to enable Monitoring or enter data that will be available from your instances once they launch.

Kernel ID: Use Default

RAM Disk ID: Use Default

Monitoring: Enable CloudWatch detailed monitoring for this instance (additional charges will apply)

User Data:

Termination Protection: Prevention against accidental termination.

Shutdown Behavior: Choose the behavior when the instance is shutdown from within the instance.

10. Deployment Stack Hardening



```
#Your Special cloud-config Script
```

```
apt_update: true
```

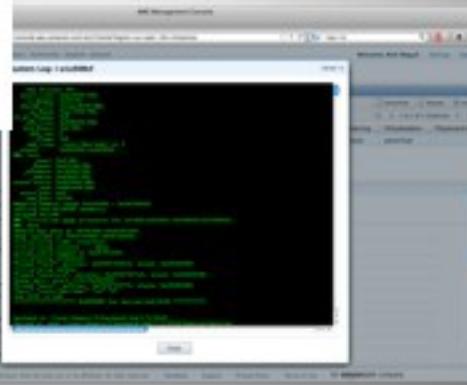
```
apt_upgrade: true
```

```
packages:
```

- cryptsetup
- xFce
- mysql-server
- php5
- php5-mysql
- curl

Automatically Patch

Automatically Upgrade



Patching

10. Deployment Stack Hardening



```
#Your Special cloud-config Script
```

```
fixrouting:
```

```
- &fix_routing |  
public_ipv4=$(curl -s http://169.254.169.254/latest/meta-data/public-ipv4)  
ifconfig eth0:0 public_ipv4 up
```

```
config(mysql):
```

```
- &config_mysql |  
/etc/init.d/mysql stop  
mkdir /encrypted  
/etc/init.d/openssl teardown  
cd /etc/openssl  
rm /etc/openssl/* /usr/share/mysql  
wget https://s3.amazonaws.com/YourCa/YourDrive/usr_share_mysql  
cd /etc/mysql  
wget https://s3.amazonaws.com/YourCa/YourDrive/csk-mysql-startup  
mv /etc/mysql/my.cnf /etc/mysql/my.cnf.bak  
wget https://s3.amazonaws.com/YourCa/YourDrive/my.cnf  
chmod 644 /etc/mysql/my.cnf  
/etc/init.d/openssl start  
rm /etc/update-motd.d/*  
cd /etc/update-motd.d  
wget https://s3.amazonaws.com/YourCa/YourDrive/98-header
```

```
runcmd:
```

```
- [ sh, -c, *fix_routing ]  
- [ sh, -c, *config_mysql ]
```

Configuration

10. Deployment Stack Hardening

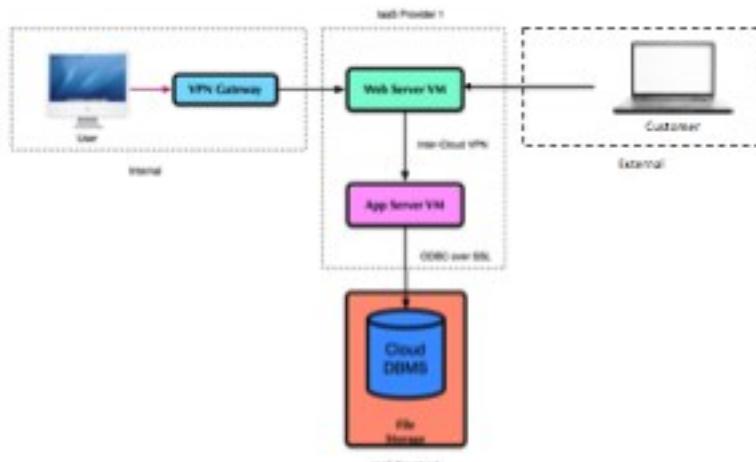


- Security zones are your basic security control
- You define access points and data flow
- Provider may not allow full visibility to networks
- Provider shields you from other customers

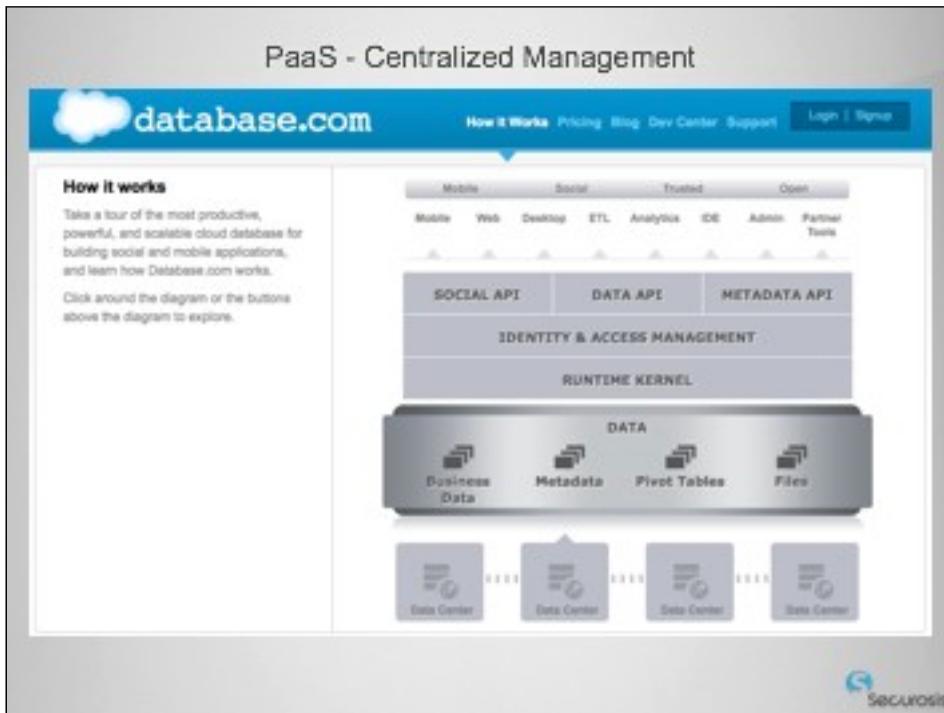
11. Security Zones



Security Zones



12. Management Plane

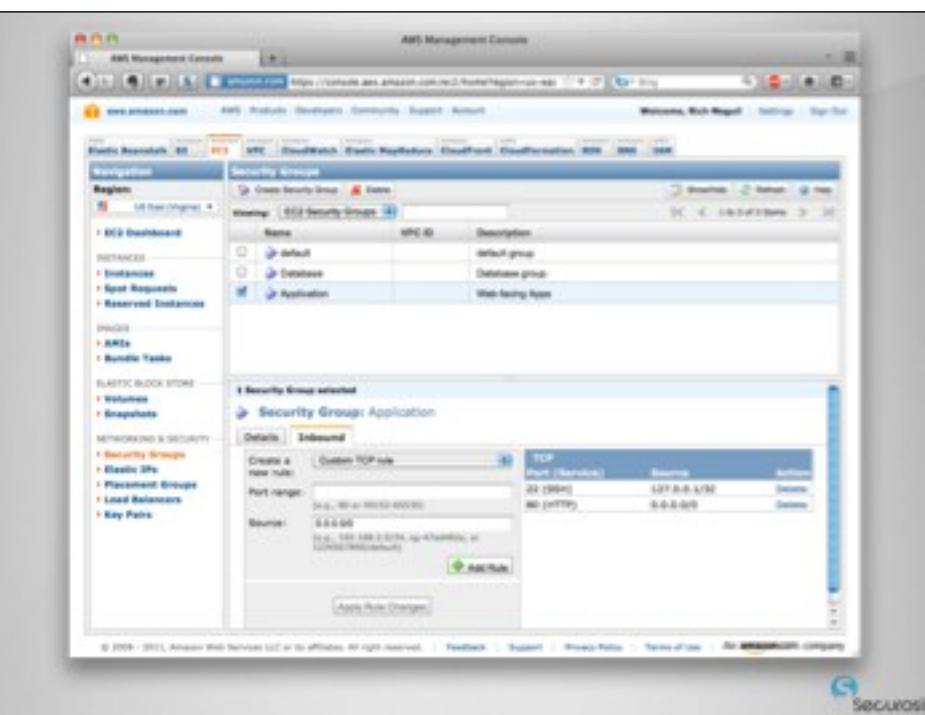


Geared for Development & Deployment



The screenshot shows the Google App Engine homepage. At the top, there's a navigation bar with links for Home, Docs, FAQ, Articles, Blog, Comments, Terms, and Download. Below the navigation is a large blue icon of a computer monitor with a brain-like swirl on its screen. The main heading is "Run your web apps on Google's infrastructure". A sub-headline says "Easy to build, easy to maintain, easy to scale". A paragraph explains that Google App Engine enables users to build and host web apps on the same systems that power Google applications. It highlights features like fast development and deployment, simple administration, and effortless scalability. A sidebar on the right contains sections for "Getting Started", "Get Involved", and "Watch and Learn", each with a small image and a brief description. The "Getting Started" section includes steps: 1. Sign up for an App Engine account, 2. Download the App Engine SDK, 3. Read the Getting Started Guide. The "Get Involved" section lists ways to contribute to the community. The "Watch and Learn" section features a video thumbnail with the text "Developing and deploying on Google App Engine. Watch Now". On the left side, there's a "Focus on your app, leave the rest to us" section with a small image of a developer and a list of benefits. Below that is a "Download" section with a green arrow pointing down, listing SDKs for Python, Java, and Go, along with links to documentation and the Google Plugin for Eclipse. A "Dive Deeper" section provides links to the App Engine blog, Go and Java documentation, and Python documentation.

SIECAVOSIS

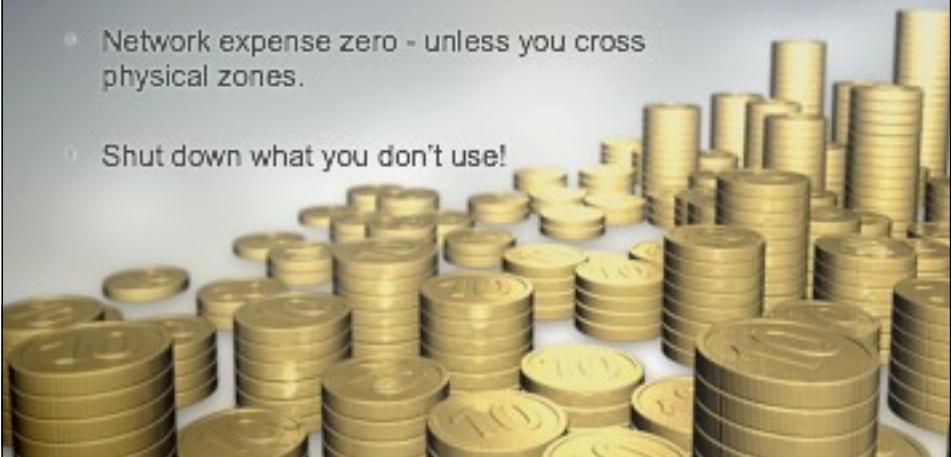


The screenshot shows the AWS Management Console interface for managing security groups. The left sidebar has navigation links for EC2 Dashboard, Instances, Spot Requests, Reserved Instances, AMIs, and Elastic Block Store. Under Network & Security, it lists Security Groups, Elastic IP, Placement Groups, Load Balancers, and Key Pairs. The main content area is titled "Security Groups" and shows a list of existing security groups: "default", "Database", and "Application". The "Application" group is selected. Below the list, a detailed view shows the "Security Group selected" for "Application". It has tabs for "Details" and "Inbound". Under "Details", there's a "Create a new rule" dropdown set to "Custom TCP rule", a "Port range" input (24-30), and a "Source" input (0.0.0.0/0). Under "Inbound", there are two entries: "TCP" (port 22) from "0.0.0.0/0" to "22 (SSH)" and "TCP" (port 80) from "0.0.0.0/0" to "80 (HTTP)". There are "Add Rule" and "Apply Rule Changes" buttons at the bottom. The URL in the browser is https://console.aws.amazon.com/ec2/home?region=us-east-1#security-groups:application.

SIECAVOSIS

Cost Considerations

- Inexpensive to design for scale
- Network expense zero - unless you cross physical zones.
- Shut down what you don't use!



Provider Issues

- API's and lock-in - abstraction more important
- Secure AMI's
- Network addressing and availability zones
- Lack of logs in multi-tenant environments
- SOD and admin access

Cloud-Sec 12-Step

Adrian Lane

Securosis, LLC

@adrianlane

alane@securosis.com

