Do They Deliver?

Practical Load and Security Testing of Cloud Service Providers

Matthias Luft
&
Enno Rey

{mluft,erey}@ernw.de

ERNW GmbH

Heidelberg based security consulting and assessment company.

- Independent
- We understand corporate
- Deep technical knowledge
- Structured (assessment) approach
- Business reasonable recommendations

- Blog: www.insinuator.net
- Conference: www.troopers.de

Agenda

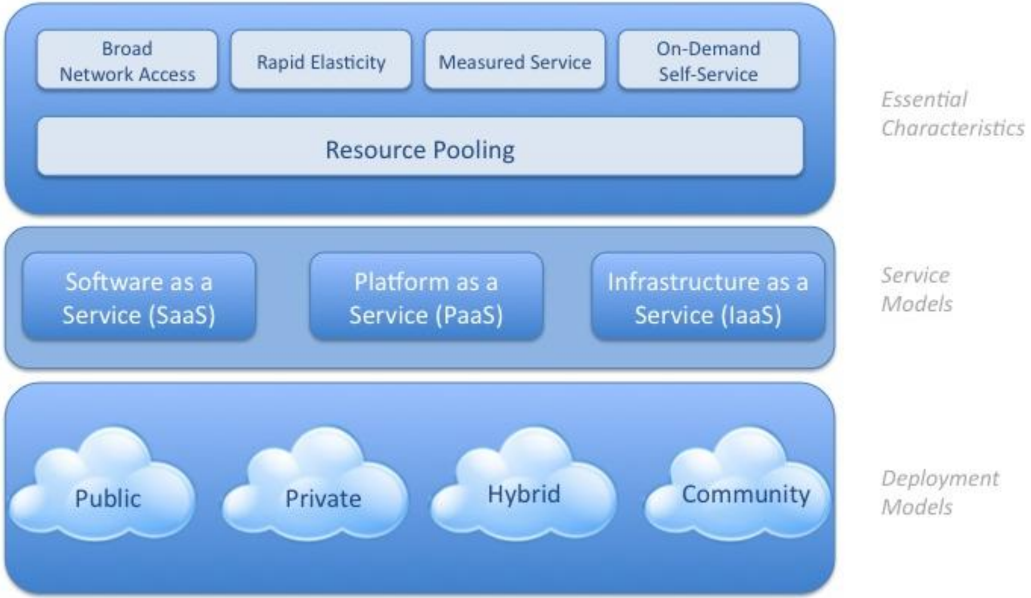
- Definition of Cloud Computing
- Motivation
- Load Testing Results
- Security Testing Results
- Conclusions

- This talk is a loose collection of “war stories” and (research) project anecdotes
  - Which reflects ERNW’s “practical approach”.
  - Still, there are quite some lessons to learn…

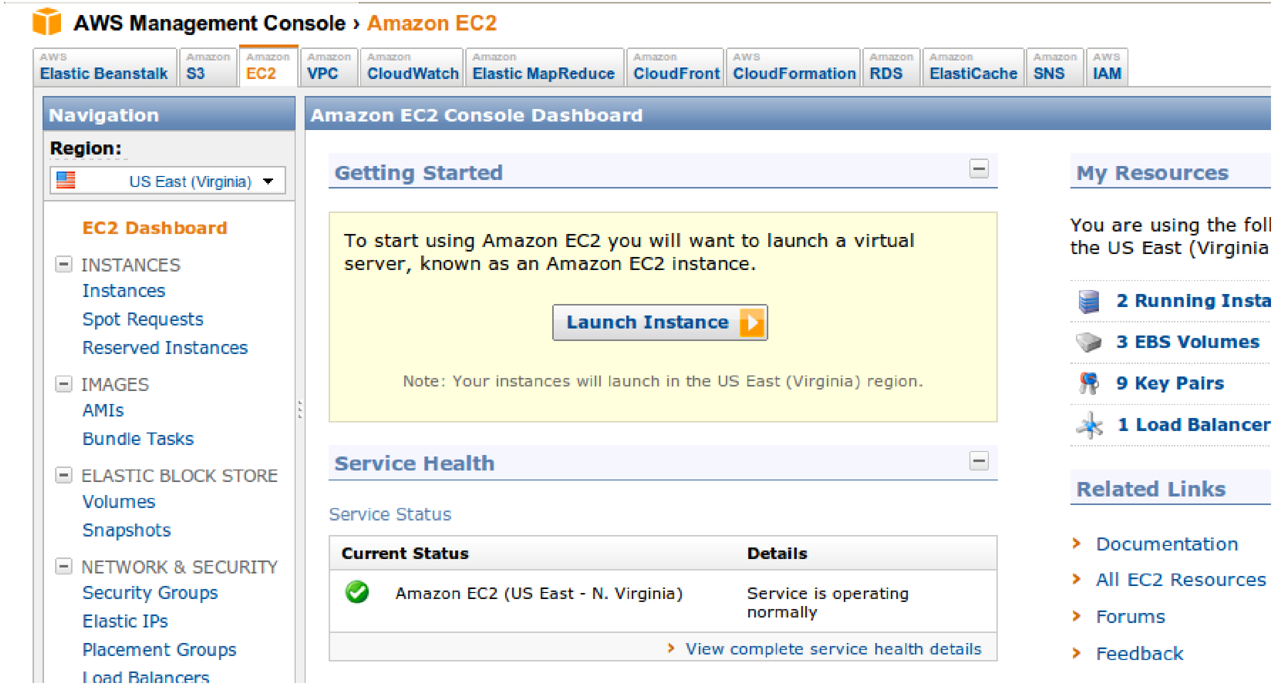
Definition

Visual Model Of NIST Working Definition Of Cloud Computing

http://www.csrc.nist.gov/groups/SNS/cloud-computing/index.html



The practical approach



Amazon EC2 Console Dashboard

Getting Started

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

Note: Your instances will launch in the US East (Virginia) region.

My Resources

You are using the foll
the US East (Virginia

2 Running Insta

3 EBS Volumes

9 Key Pairs

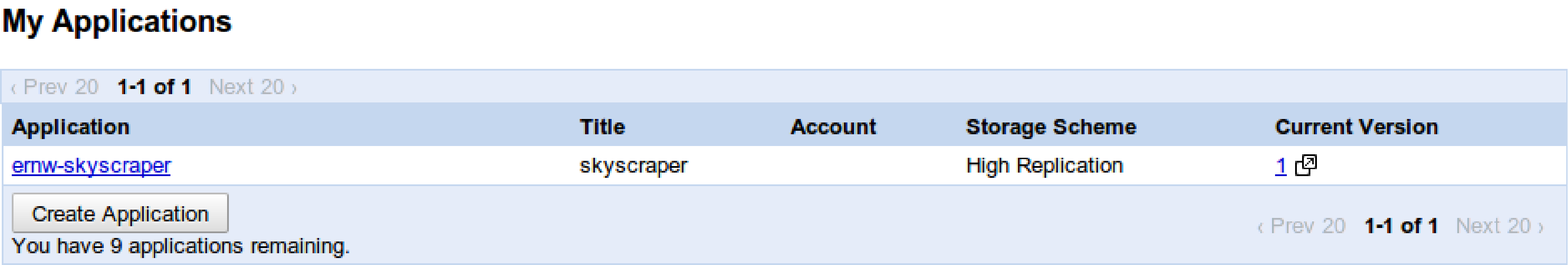
1 Load Balancer

Service Health

Service Status

Current Status	Details
✓ Amazon EC2 (US East - N. Virginia)	Service is operating normally

Practical Approach (II)



My Applications

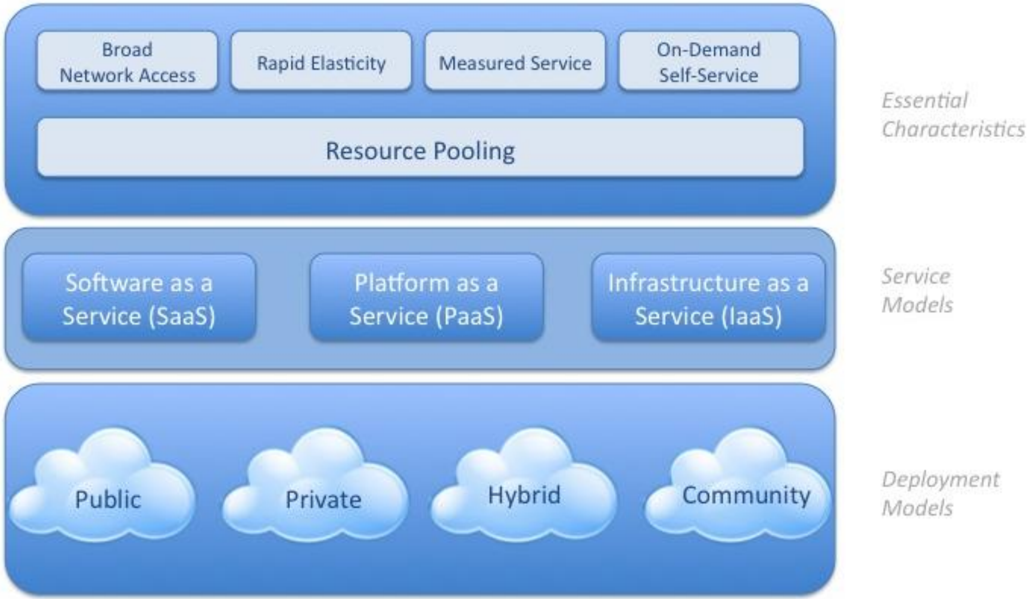
Application	Title	Account	Storage Scheme	Current Version
ernw-skyscraper	skyscraper		High Replication	1 ⧉

You have 9 applications remaining.

Definition

Visual Model Of NIST Working Definition Of Cloud Computing

http://www.csrc.nist.gov/groups/SNS/cloud-computing/index.html



Broad Network Access

Rapid Elasticity

Measured Service

On-Demand Self-Service

Resource Pooling

Essential Characteristics

Software as a Service (SaaS)

Platform as a Service (PaaS)

Infrastructure as a Service (IaaS)

Service Models

Public

Private

Hybrid

Community

Deployment Models

Motivation

- Emerging new technology
- New computing paradigm
- Decreased costs
- Improved availability

So…

¬ Do they deliver?

Load Testing

Do they deliver?

- Do Cloud Service Providers (CSPs) deliver scalable environments?
  - ➢ Performance evaluation methodology for *PaaS* environments!

- Methodology:
  - – Development of a sample application
  - – Following platform scalability best practices
  - – Load testing of three different PaaS environments:
    - – Google App Engine
    - – Amazon Elastic Beanstalk
    - – Salesforce Force.com
  - – Comparison to the *old world*: load test of a physical server

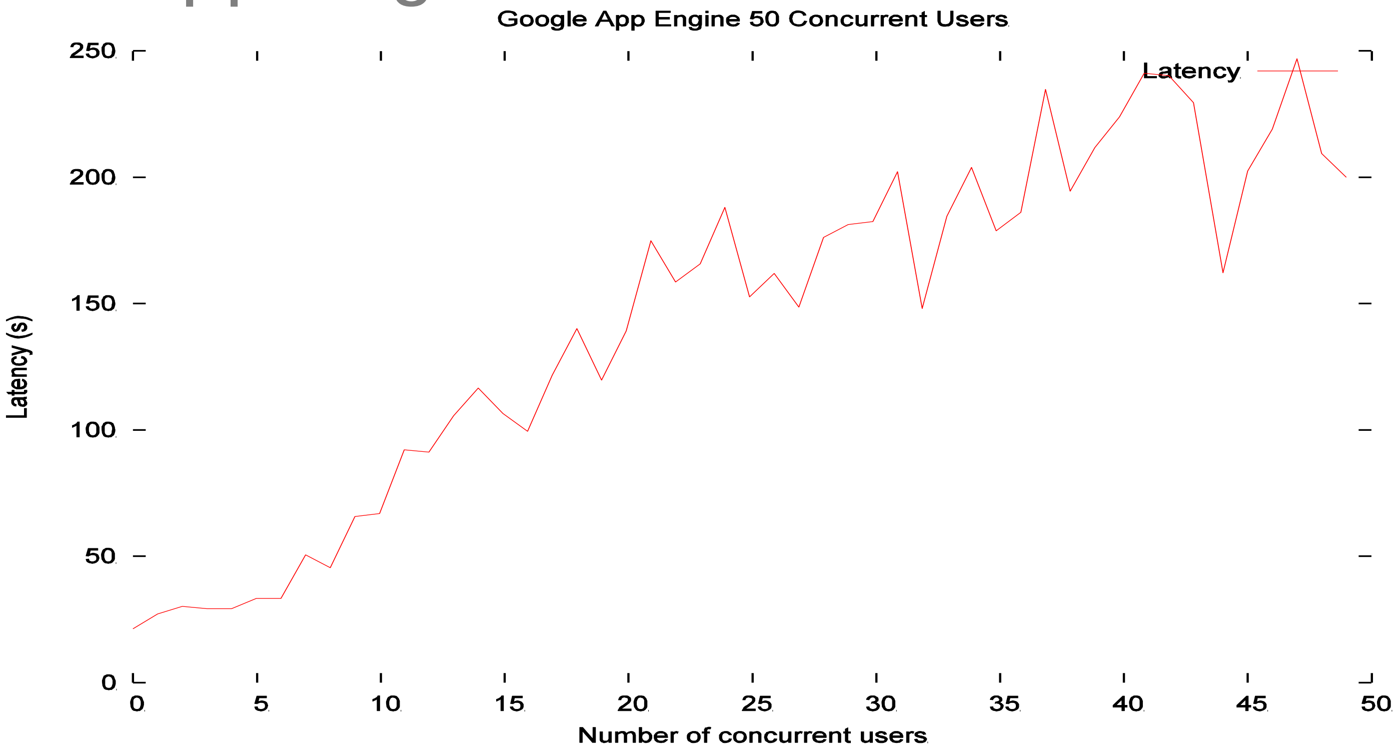
Tasks & Challenges

- Implement application on any platform
- Load on any component of the PaaS stack
- Follow best practices for different platforms
- JavaScript/Load test friendly UI
- Stable load testing framework

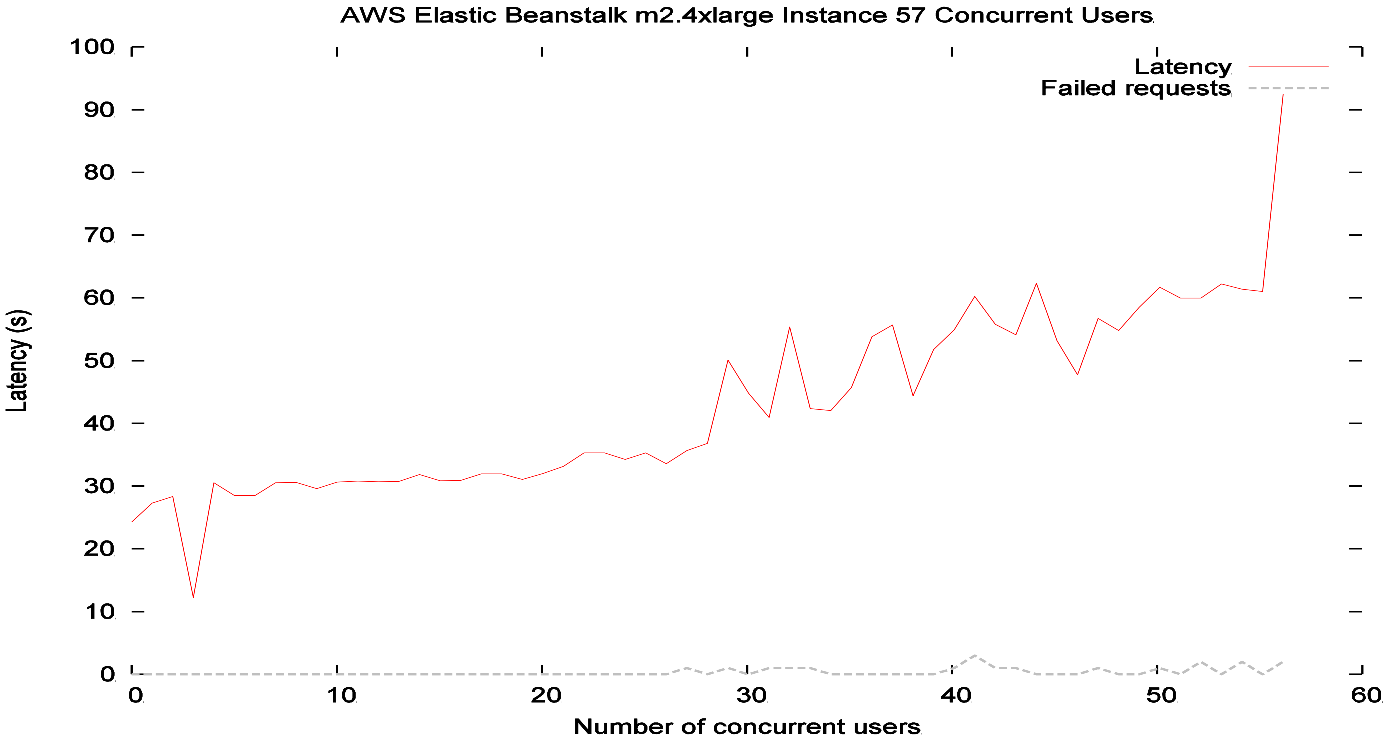
Results

- Latency
- Increasing number of concurrent users
- Number of failed/timed out requests

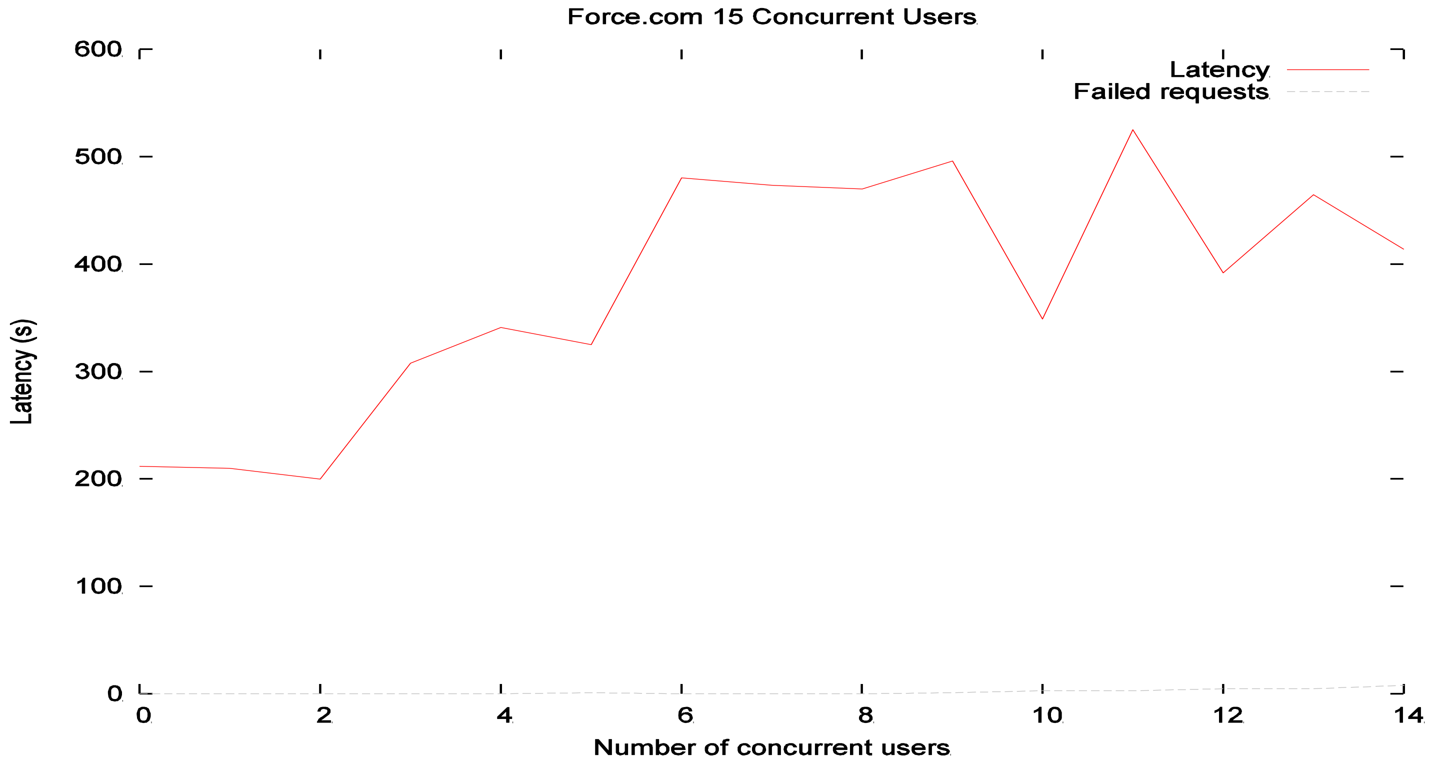
Results App Engine



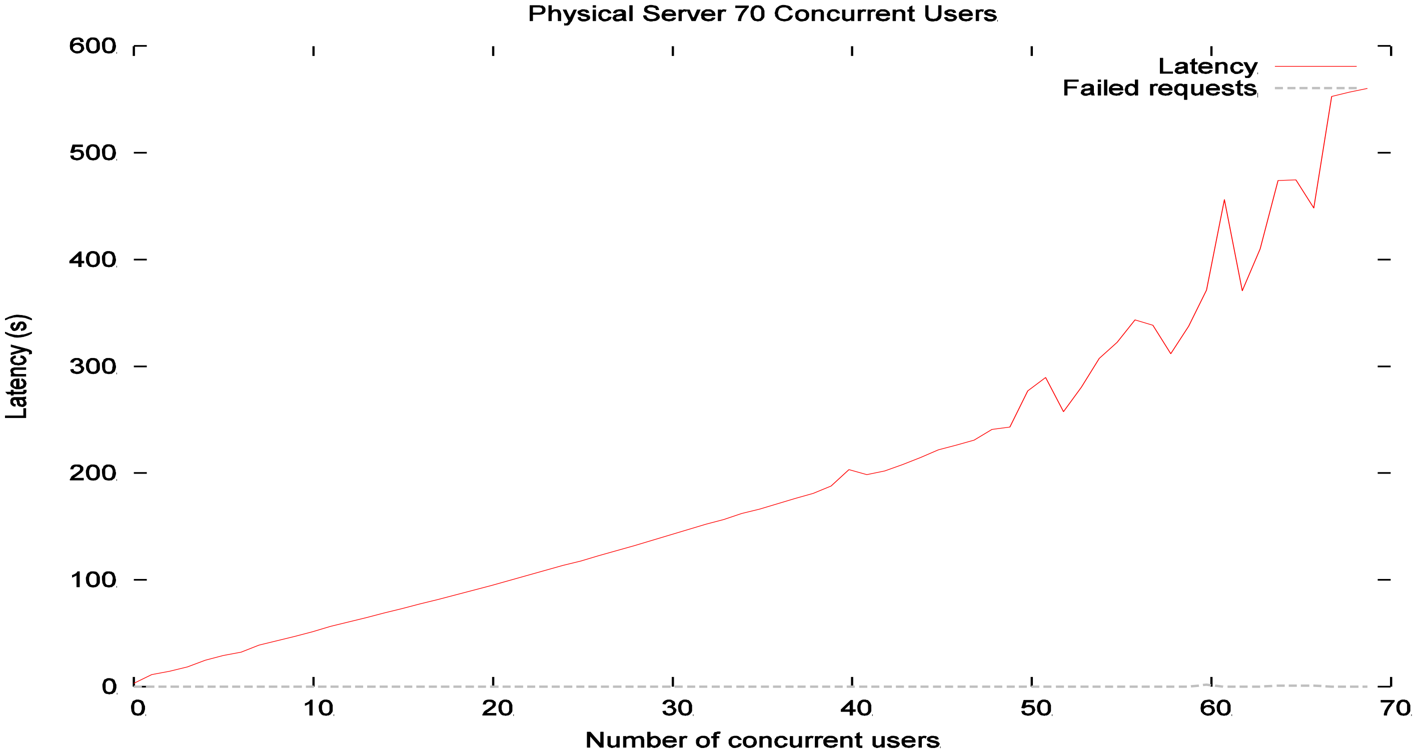
Results Elastic Beanstalk



Results Force.com



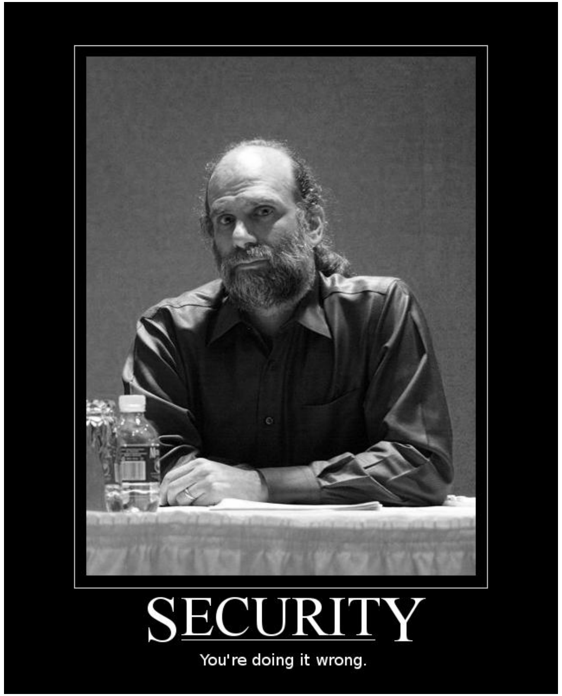
Results Physical Server



Results Summary

- Each CSP exhibits different characteristics
  - Further research for JS based UIs necessary
- Virtualization performance issues (?)
- Horizontal vs. vertical scalability
- So, the basic idea still sounds pretty nice?
  - Cloud fanboy? Remember the Amazon example? ;-)

Security



- *Security Concerns in Cloud Computing Environments*?
- Rather general statement!
  - The concerns about "security concerns" do not allow a structured discussion…
- Let's have a look at an example…

Security

You’re doing it wrong.

- Evaluation of a SaaS CSP
  - Some “HR management software”
- They agreed to perform a pentest on behalf of the potential customer.
  - Which is not necessarily the case!
  - See the next slides.

Pentesting SaaS

- Target of evaluation: HR web application
- First approach: Typical web application pentest
- “Cloud testing approach”: Not necessary….

Pentesting SaaS

- Basic result: After one day, we stopped the test.
  - We already had more severe findings than in some other 20 man day tests ;-)

- So, no need to test cloud related stuff…

Pentesting SaaS

- *HTTP PUT* to the web root was possible!
- Seriously, when did you see something like that the last time?
- Possible reasons:
  - Short time to market
  - Abstraction from technical concepts
  - Platform focus on easy implementation

Taking the structured approach

Again, this stays in the realm of theory ;-)

- System operation life cycle
- Risk analysis
- Trust evaluation
- Contracts, questionnaires & audits
  - Does this really work?

Introducing the System Operation Life Cycle

1. Hardware is purchased. . .
2. . . . from trusted hardware suppliers.
3. The hardware is operated in own data centers. . .
4. . . . which reside in carefully selected countries and locations. . .
5. . . . and are secured by carefully selected access control mechanisms.
6. The hardware is operated by trusted employees. . .
7. . . . who install operating systems. . .
8. . . . from trusted install media. . .
9. . . . in a secure, documented way.
10. The operating system is secured by carefully selected controls.
11. Only approved applications are installed. . .
12. . . . from trusted install media. . .
13. . . . and operated and secured using carefully developed guidelines.
14. Hosted applications are developed following carefully developed secure coding guidelines.

Implications

of the Life Cycle

- The System Operation Life Cycle depicts areas of control.
- Securing the cloud?
  - Regarding the life cycle, hardly possible
  - Also not desired!
- There are several risks that are intrinsic to cloud computing environments.

Risks

- High risks according to ENISA:

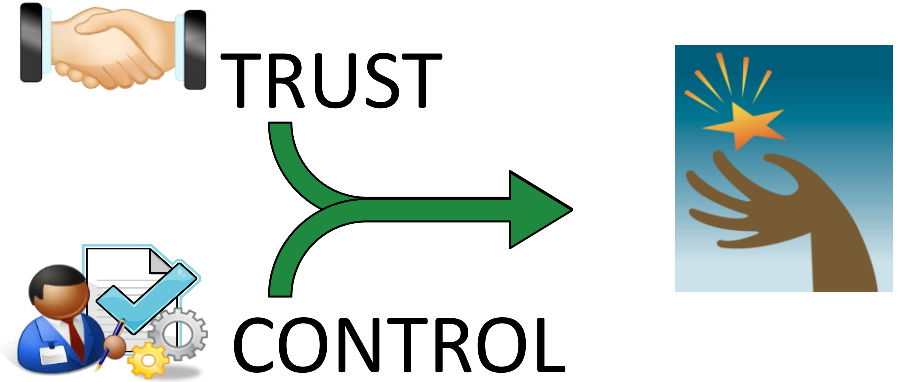
- Lock-In
- Loss of Governance
- Compliance Challenges
- Isolation Failure
- Cloud Provider Malicious Insider – Abuse of High Privilege Roles
- Subpoena and E-Discovery
- Risk from Changes in Jurisdiction
- Data Protection Risks

Risks & The Life Cycle

- Loss of governance:
  - Steps in the life cycle: 1-13
- Isolation Failure:
  - Steps in the life cycle: 3-13
- Malicious Insider
  - Steps in the life cycle: 1-6

Trust and Control

- Lack of technical controls
- => Qualified trust for secure operation!



TRUST

CONTROL

Trust Factors…

- Size
- *Symmetry*
- *Transparency*
- *Consistency*
- Integrity
- *Value of Reward*
- Components
- Porosity

… applied to CSPs

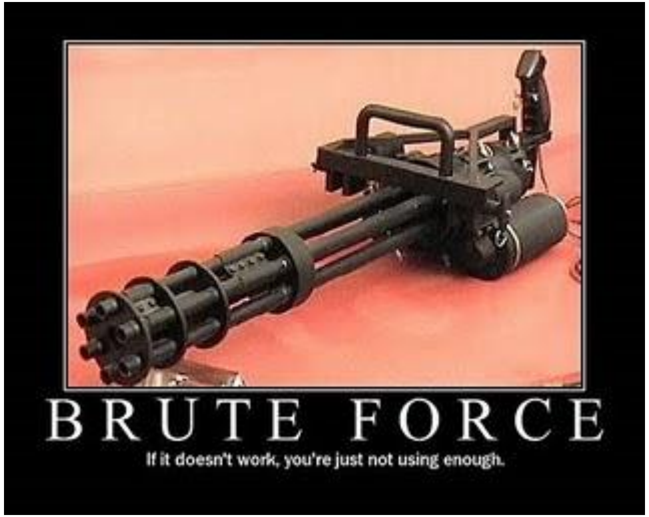
Based on the evaluations.

- Symmetry & Transparency
  - Trust to perform a pentest
- Consistency
  - Well, based on the first results… ;-)
- Value of Reward
  - Can be partly evaluated by the proposed load test methodology

Ongoing improvement

- Since many customers are interested in Amazon as a CSP, we perform a lot of tasks in the Amazon cloud.
- In the course of one of our regular password audits, we discovered some abnormalities in the Amazon login procedure.
  - **Drop that, we wanted to break that stuff ;-)**

Amazon Bruteforce



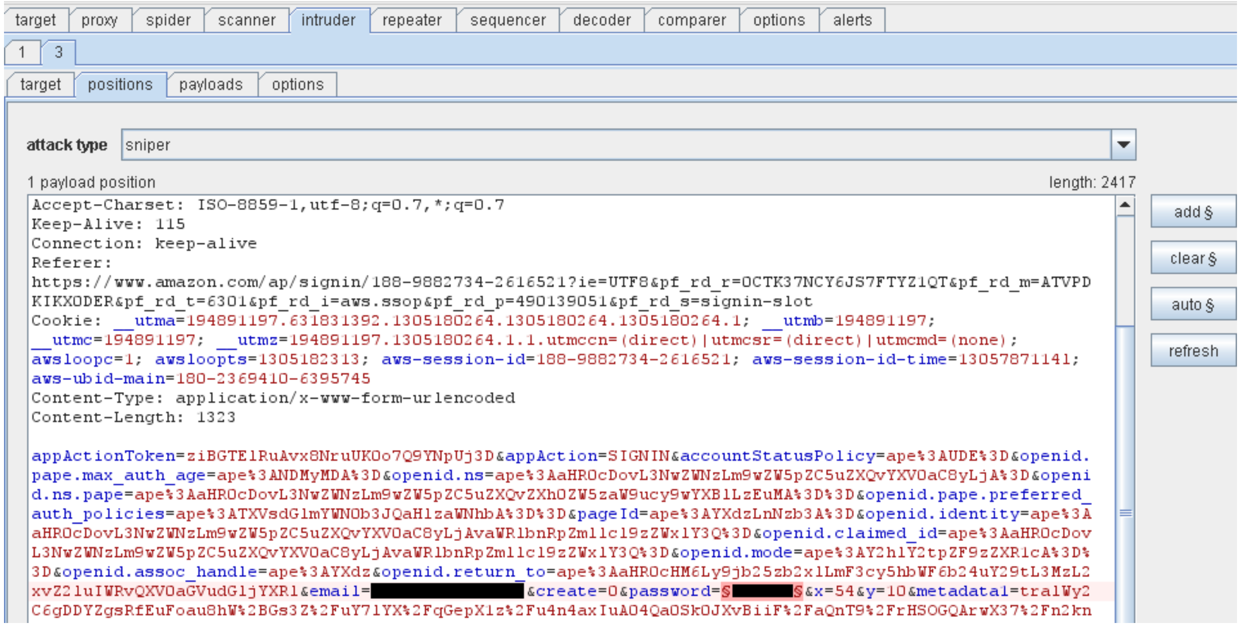
- Password bruteforcing attempt
- Using our own account
- Using the standard Amazon login form

Amazon Bruteforce

- Tricky setup
- Most bruteforcing tools do not cope well with modern webapp authentication mechanisms
  - Cookies with different scopes, redirects, JavaScript

Amazon Bruteforce

- Using *Burp* for the bruteforcing



attack type: sniper

1 payload position   length: 2417

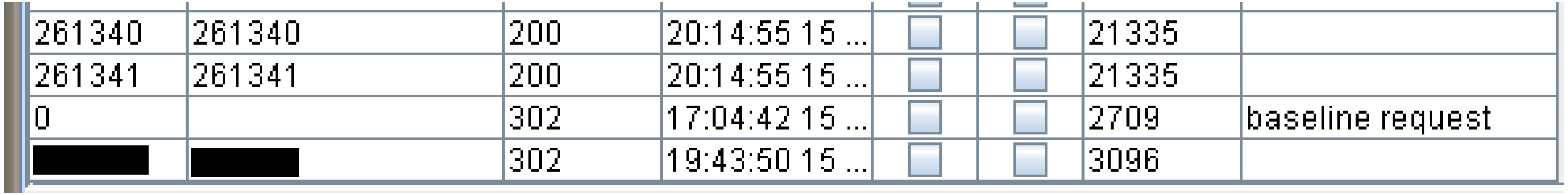
```
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Connection: keep-alive
Referer:
https://www.amazon.com/ap/signin/188-9882734-2616521?ie=UTF8&pf_rd_r=0CTK37NCY6JS7FTYZ1QT&pf_rd_m=ATVPD
KIKX0DER&pf_rd_t=6301&pf_rd_i=aws.ssop&pf_rd_p=490139051&pf_rd_s=signin-slot
Cookie: __utma=194891197.631831392.1305180264.1305180264.1305180264.1; __utmb=194891197;
__utmc=194891197; __utmz=194891197.1305180264.1.1.utmccn=(direct)|utmcsr=(direct)|utmcmd=(none);
awsloopc=1; awsloopts=1305182313; aws-session-id=188-9882734-2616521; aws-session-id-time=13057871141;
aws-ubid-main=180-2369410-6395745
Content-Type: application/x-www-form-urlencoded
Content-Length: 1323

appActionToken=ziBGTElRuAvx8NruUK0o7Q9YNpUj3D&appAction=SIGNIN&accountStatusPolicy=ape%3AUDE%3D&openid.
pape.max_auth_age=ape%3ANDMyMDA%3D&openid.ns=ape%3AaHR0cDovL3NwZWNzLm9wZW5pZC5uZXQvYXV0aC8yLjA%3D&openi
d.ns.pape=ape%3AaHR0cDovL3NwZWNzLm9wZW5pZC5uZXQvZXh0ZW5zaW9ucy9wYXBlLzEuMA%3D%3D&openid.pape.preferred_
auth_policies=ape%3ATXVsdGlmYWN0b3JQaHlzaWNhbA%3D%3D&pageId=ape%3AYXdzLnNzb3A%3D&openid.identity=ape%3A
aHR0cDovL3NwZWNzLm9wZW5pZC5uZXQvYXV0aC8yLjAvaWRlbnRpZmllcl9zZWxlY3Q%3D&openid.claimed_id=ape%3AaHR0cDov
L3NwZWNzLm9wZW5pZC5uZXQvYXV0aC8yLjAvaWRlbnRpZmllcl9zZWxlY3Q%3D&openid.mode=ape%3AY2hlY2tpZF9zZXR1cA%3D%
3D&openid.assoc_handle=ape%3AYXdz&openid.return_to=ape%3AaHR0cHM6Ly9jb25zb2xlLmF3cy5hbWF6b24uY29tL3MzL2
xvZ2luIWRvQXV0aGVudGljYXRl&email=[redacted]&create=0&password=§[redacted]§&x=54&y=10&metadata1=tralWy2
C6gDDYZgsRfEuFoau8hW%2BGs3Z%2FuY7lYX%2FqGepX1z%2Fu4n4axIuA04Qa0Sk0JXvBiiF%2FaQnT9%2FrHSOGQArwX37%2Fn2kn
```

Amazon Bruteforce

- Burp might not be the best choice for bruteforcing.
- Still, good performance
  - ~80k requests per hour
- Setup was implemented in ~20 minutes
  - More details can be found in our blog www.insinuator.net.

Amazon Bruteforce

- Successful login:



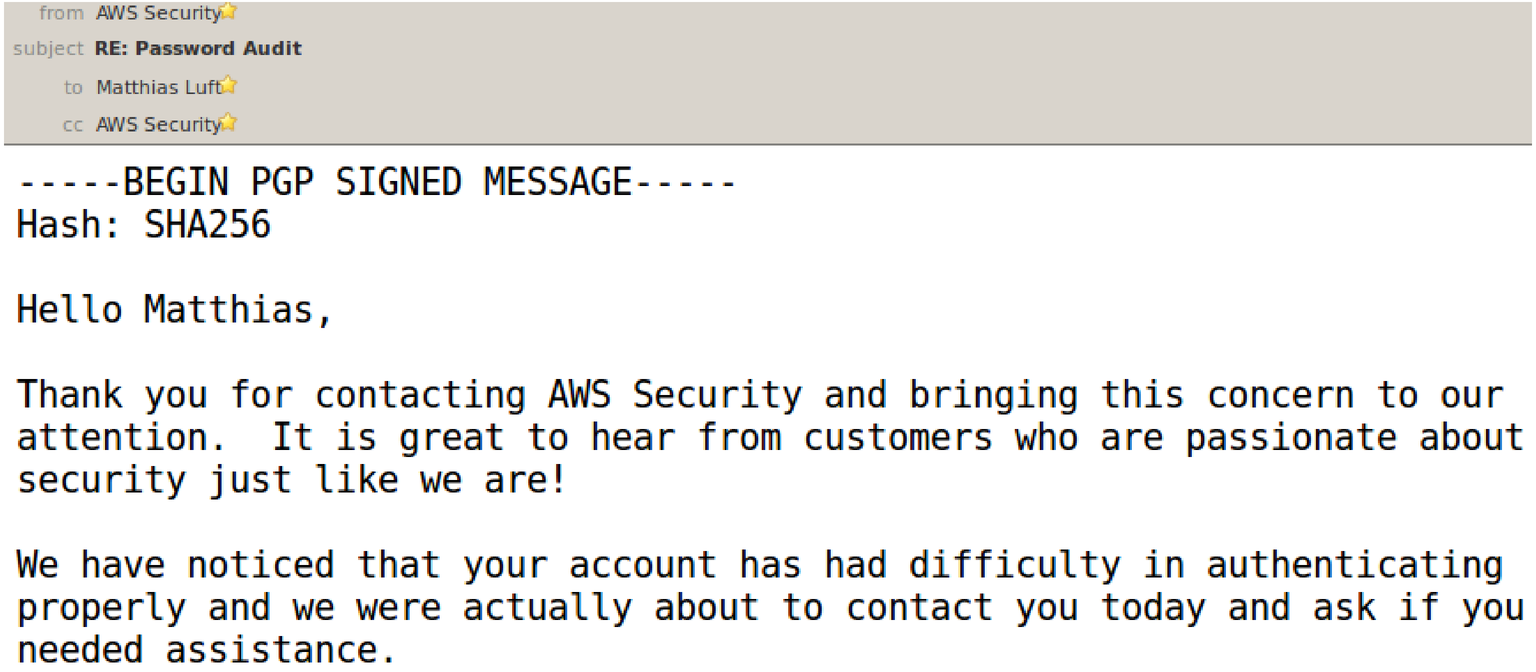
261340	261340	200	20:14:55 15 ...	☐	☐	21335	
261341	261341	200	20:14:55 15 ...	☐	☐	21335	
0		302	17:04:42 15 ...	☐	☐	2709	baseline request
[redacted]	[redacted]	302	19:43:50 15 ...	☐	☐	3096	

Amazon Bruteforce

- Bruteforcing is possible. Big surprise?
- More important:
  - No connection throttling!
  - No account lockout!
  - No captcha solution!

Amazon Bruteforce

- Following our usual responsible disclosure approach, we contacted Amazon…



from AWS Security
subject **RE: Password Audit**
to Matthias Luft
cc AWS Security

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA256

Hello Matthias,

Thank you for contacting AWS Security and bringing this concern to our
attention.  It is great to hear from customers who are passionate about
security just like we are!

We have noticed that your account has had difficulty in authenticating
properly and we were actually about to contact you today and ask if you
needed assistance.
```

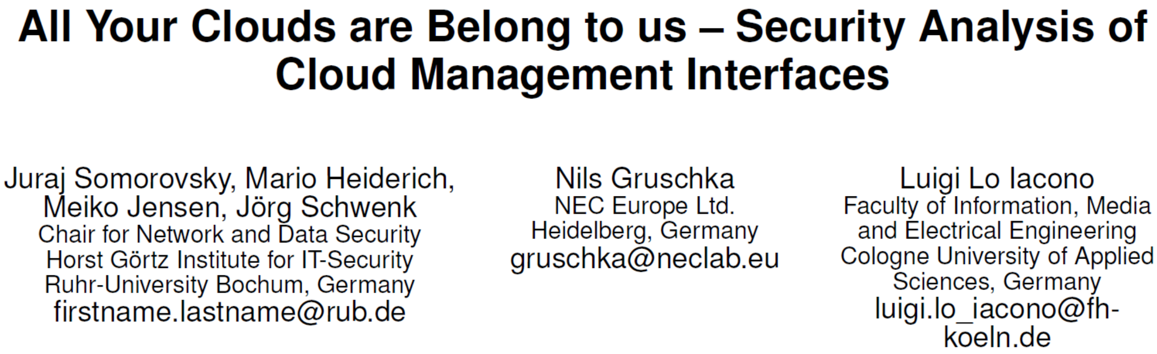
Amazon Aftermath

- Very good response of the Amazon Security Team!
- Fast implementation of a captcha solution.
- Re-evaluation of our bruteforce attempt.

The day after...

some paper gained attention, which was presented at the

CCSW 2011: The ACM Cloud Computing Security Workshop



All Your Clouds are Belong to us – Security Analysis of Cloud Management Interfaces

Juraj Somorovsky, Mario Heiderich, Meiko Jensen, Jörg Schwenk
Chair for Network and Data Security
Horst Görtz Institute for IT-Security
Ruhr-University Bochum, Germany
firstname.lastname@rub.de

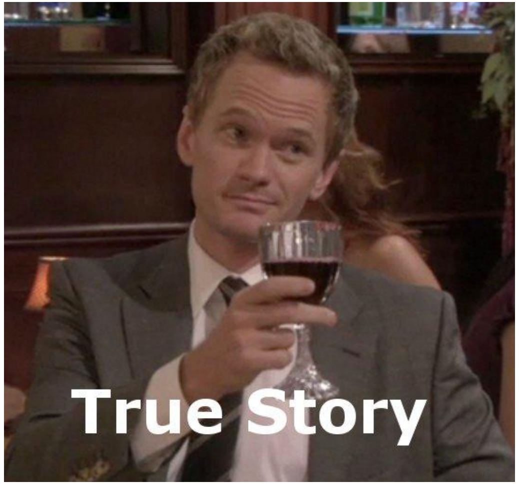
Nils Gruschka
NEC Europe Ltd.
Heidelberg, Germany
gruschka@neclab.eu

Luigi Lo Iacono
Faculty of Information, Media and Electrical Engineering
Cologne University of Applied Sciences, Germany
luigi.lo_iacono@fh-koeln.de

General Conclusion

- Web interfaces are now the *key to our data centers*.
- Overall security posture of web applications?
  - Remember the *HTTP PUT* stuff? ;)
- Maturity of the **Amazon**(!!!) web interface?

Summary



- Practical testing might lead to interesting results ;-)
- Understand the scalability approach your favorite CSP offers
  - And develop (& load test) accordingly.
- Protect the management interfaces!

Thank you for your attention!

Questions?

References

- http://www.nds.rub.de/media/nds/veroeffentlichungen/2011/10/22/AmazonSignatureWrapping.pdf