Deployments in PCI-DSS environment

Máté Marjai - SumUp

Introduction

What is SumUp?

Why PCI-DSS is important to us?

Payments Card Industry Data Security Standard

Hardware restrictions

Access restrictions (virtual and physical)

Logs, logs, logs

Encryption (Network, DB)

Regular scans and reviews

DMZ over servers within

Network restrictions

No scripts, dev libs, compilers, source control

Card details - Dos

Encrypt PAN (Primary Account Number)
Mask PAN
Cardholder name
Last 4 digits

Card details - Don'ts

Full mag-stripe data
Clear text PAN
CVV
PIN
BIN + last 4 digits

SumUp's setup

- Using a PCI-DSS cloud provider - offloads hardware reviews
- Card details and payment processing in DMZ
- Everything else outside DMZ

Deployment procedures

Segregation of duties
Software dev guidelines (code reviews)
Temp access to servers on deployments
Restricted access to DB
Read-only access

Conquering the constraints

0th Iteration:

- Shell scripts and manual deploys
- scp tar.gz files; extract; symlink; restart web server...

Conquering the constraints

1st Iteration:

- Using Capistrano for deployments
- Assets precompile and gem bundles on a dedicated server
- Running migrations on dedicated server only
- Manual run on unit tests and acceptance test suite

Conquering the constraints

2nd Iteration:

- Making builds using Jenkins (assets, gems, unit tests, etc...)
- Using these builds on deployments with Capistrano
- Trigger builds with github commit + push
- Tag successful results on github

What's next?

Introduce CFEngine for server configs

Cluster cfg clones

CI (unit tests, BDD, etc...)

On-Demand deployments from Jenkins to Staging/Production

Automatic deployments on successful builds to test servers

Tie Jenkins with ticketing system

BOW