Security, accessibility (508) and change management -

What we've learned as managers and developers

Linda Newman (newmanld@uc.edu),
Glen Horton (hortongn@uc.edu),
Thomas Scherz (scherztc@uc.edu)

2015-09-22

https://scholar.uc.edu

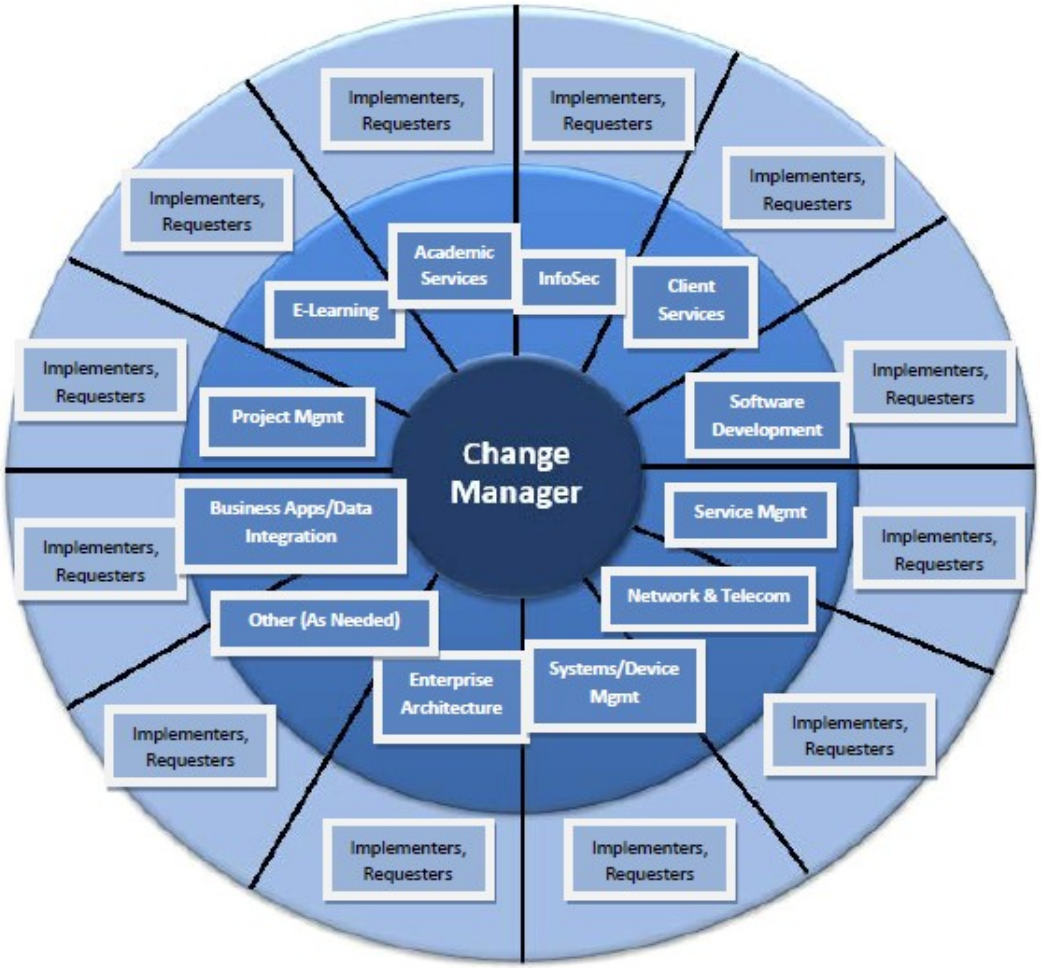

Illustration from University of Cincinnati Information Technologies, *Change Management Process*, Revised 10/1/14 BCT

Change Management at our institution is the process of planning, coordinating, implementing and monitoring changes affecting any production platform within Information Technology's control. Changes can be anything from re-routing cables to deploying code.

UNIVERSITY OF Cincinnati

Typical examples on the CAB agenda:

5144 – ESS – 11/28/14 – 4:00am – (Scheduled – Moderate Risk) PAPA - front door router – Need to reboot PAPA to upgrade to newer code. It is equipped with redundant supervisor cards, they will be rebooted separately to minimize any interruption of service. Each card will take approximately 10 minutes to reboot. Since they are redundant, user traffic should not be affected.

5132 – ESS – 11/29/14 – 8:00am – (Scheduled – Low Risk) bbservices database and web service - We will be shutting down IIS and the MS SQL database so that the SA group can get a good backup of the system. This will require downtime for the sysop tool and course eval feed generator. I have verified with Lisa and Brenda that this is ok with them. – 2 hours

5963 – ESS – 9/3/2015 – 8:00am – (Scheduled – Low Risk) Data Domain (DD990) – Disable the 10.23.15.143 interface on DD990. This is currently configured as a 3 port LACP group and is no longer needed. All services have been converted to use a 10Gb interface. This CM will also cover the removal of the physical cabling and network configuration. No downtime required. – 1 hour

Before we submit a Change Request, we complete an assessment.

Security Level Assessment

SLA Attributes	1	2	3	4
Data Type	Restricted Data	Controlled Data	Private Data	Public Data
System/Service Type	Restricted Web Based System or Application	Publicly Accessible Web Based System / Application / Site	Services or Components accessible via external network	Services or Components accessible only via internal network
Change Type	Major Modification to core components – includes patches and upgrades	Moderate to Minor modifications to core components	Modifications to UI components that include JavaScript or JQuery modifications. Database level modifications – Restricted to procedures/queries that include inserts/update/deletes	UI modifications restricted to CSS and HTML changes. Database level modifications – Restricted to procedures/queries focus that export or produce data views (no inserts/update/deletes)
Users Affected	Entire Organization	One or More Colleges	Select departments, units and/or groups	Individual or Small Group

Security Risk Level	Risk
4-8	High
9-12	Medium
13-16	Low

Any Medium or High risk will likely trigger a request to repeat our security scan.

UNIVERSITY OF Cincinnati

Photograph: Amelia's Sad Face, https://www.flickr.com/photos/donnieray/9436653177 licensed as https://creativecommons.org/licenses/by/2.0/

This is challenging for agile.

As one expert wrote "formal change management often evolves into a change prevention strategy on the part of IT staff."

(http://www.drdobbs.com/architecture-and-design/discip)

All is NOT bleak.

- Our Quality Assurance team developed enough confidence in us to allow us to run the security scan ourselves – probably the only way we could get it done as you will hear about shortly.
- We're talking with the head of Change Management and the lead of the other agile project about more pragmatic approaches.
- We're now moving on to accessibility.

To better optimize our change management process for agile, we are looking for help in these areas:

- Inline security tools that can run when we deploy code, much like Travis.
- Involvement of our QA and Information Security staff in reviewing our actual code, not just checking off the results of a software tool.
- Advice from you!

Security Scanning

Trustwave's Hailstorm

Part of change management

zero harm score required for deploys

45,714 attacks over 14 hours

QA/Staging Environment

(matches production environment)

Hailstorm = extreme stress test

Fedora 3 memory leaks

more CPUs and RAM for Fedora/Solr

Hailstorm Requirements

Redirect all 500 errors to simple 404s

limit every passed parameter

no integer/buffer overflows

no XSS or blind SQL injections

Frustrations

one big queue

custom app traversals

attacks can vary with each run

deployments & scanning are too slow

frequent need to wipe data

What did Hailstorm find?

Cross - Site Scripting

Cross-Site Scripting vulnerability found
Injected item: GET: works
Injection value: 1 src=javascript:alert(14404350.2827)
Detection value: 1 src=javascript:alert(14404350.2827)

Application Exceptions

Possible application exception vulnerability.
Server response status: 500
Internal server error
Injectable request #: 15
Injected item: POST: hydramata_group
%5Bmembers_attributes%5D%5B1%5D%5Bid%5D
Injection value: %00

Buffer Overflows

Buffer overflow vulnerability found
Injectable request #: 75
Injected item: POST: image%5Brelated_work_tokens%5D
Injection length: 4135

UNIVERSITY OF Cincinnati

Blind SQL Injections

Blind SQL Injection vulnerability found
Injectable request #: 64
Injected item: POST: image%5Bsource%5D%5B%5D
Light Injection value :' waitfor delay'0:0:5'--
Heavy Injection value :' waitfor delay'0:0:10'--
Detection value: <TIME BASED SQL INJECTION DETECTED>
Response Time for empty injected query = 61255ms
Response Time for light injected query = 109853ms
Response Time for heavy injected query = 175543ms

Invalid Byte Sequence in UTF-8

Possible application exception vulnerability.
Server response status: 500
Internal server error
Injectable request #: 5
Injected item: GET: works
Injection value: %84

Types of Resolutions

Error Page Sanitizing

Exception Trapping (Controllers):

```
unless Rails.application.config.consider_all_requests_local
   rescue_from Exception, with: :render_404
   rescue_from ActionController::RoutingError, with: :render_404
   rescue_from ActionController::UnknownController, with: :render_404
   rescue_from ActiveRecord::RecordNotFound, with: :render_404
 end
```

Request Handling (Middleware):

```
SANITIZE_ENV_KEYS = %w(
   HTTP_REFERER
   PATH_INFO
   REQUEST_URI
   REQUEST_PATH
   QUERY_STRING
 )


valid = URI.decode(string).force_encoding('UTF-8').valid_encoding?
```

Parameter Validation (Helpers):

```
def limit_param_length(parameter, length_limit)
    render(:file => 'public/404.html', :status => 404, :layout => false) unless parameter.to_s.length < length_limit
end
```

UNIVERSITY OF Cincinnati

Strategies Applied

Cross- Site Scripting →	Parameter Validation
JavaScript Injections →	Parameter Validation
UTF8 Encoding →	Request Handling
Invalid Error Pages →	Error Page Sanitizing
Application Exceptions →	Exception Trapping
Buffer Overflows →	Parameter Validation
Web Server Vulnerabilities →	Apache Updates

?s

Summary of our code changes:

http://bit.ly/1Oi1sZd

Linda Newman (newmanld@uc.edu),
Glen Horton (hortongn@uc.edu),
Thomas Scherz (scherztc@uc.edu)

2015-09-22

UNIVERSITY OF Cincinnati