Keepers of the Machines: Examining How Sys Admins Manage Software Updates

Frank Li
Nathan Malkin
Lisa Rogers
Arunesh Mathur
Marshini Chetty
URGENT: CRITICAL UPDATE AVAILABLE!

DETAILS: FIXES AN ISSUE THAT WAS CAUSING RANDOM LAPTOP ELECTRICAL FIRES.

(THIS UPDATE WILL REQUIRE RESTARTING YOUR COMPUTER.)

REMIND ME LATER

Source: https://www.xkcd.com/1328/
Failure to patch two-month-old bug led to massive Equifax breach

Critical Apache Struts bug was fixed in March. In May, it bit ~143 million US consumers.

Drupalgeddon 2 wreaking havoc on 900+ sites because IT still hasn't applied updates

By Brandon Vigliarolo in Security
on June 7, 2018, 7:38 AM PST

Despite the fact that the Drupal exploit was reported-and patched-in March 2018, some 115,000 websites are still vulnerable.

Large Insurance Company Settles for $5.5 Million over "Failed To Patch" Data Breach

A large insurance company (Nationwide) agreed to pay a total of $5.5 Million to settle charges brought by 32 states resulting from the loss of critical consumer information attributable to a criminal data breach.
Prior Work: End User Patching

Out of the Loop: How Automated Software Updates Cause Unintended Security Consequences

Rick Wash, Emilee Rader, and Rick Wash
Department of Telecommunication, Information Studies, and Media

Betrayed By Updates: How Negative Experiences Affect Future Security

Kami Vaniea, Emilee Rader, Rick Wash
Department of Telecommunication, Information Studies, and Media

Tales of Software Updates: The process of updating software

Kami Vaniea
The University of Edinburgh

“They Keep Coming Back Like Zombies”: Improving Software Updating Interfaces

Arunesh Mathur, Josefine Engel, and Sonam Sobti

Quantifying Users’ Beliefs about Software Updates

Arunesh Mathur*, Nathan Malkin†, Marian Harbach‡, Eyal Peer§ and Serge Egelman††

*Princeton University
††University of California, Berkeley
‡International Computer Science Institute
§Consumers Behavioral Insights Lab, Bar-Ilan University, Israel
amathur@cs.princeton.edu nmalkin@cs.berkeley.edu mharbach@icsi.berkeley.edu
eyal.peer@biu.ac.il egelman@cs.berkeley.edu

sonam.sobti9@gmail.com

Shini Chetty
shini@umd.edu

Information Studies
College Park
Prior Work: End User Patching

Out of the Loop: How Automated Software Updates Cause Unintended Security Consequences

Rick Wash, Eric Chaline, and Tavis Ormandy

[wash, chaline, ormandy]

Betrayed By Updates: How Negative Experiences Affect Future Security

Yasmeen Rashidi

MediaLab

Software Updating Interfaces: Improving the User Experience

Arinesh Mathur

Josefine Engel

Sonam Sobti

Arinesh Mathur*, Nathan Malkin†, Marian Harbach‡, Eyal Peer§ and Serge Egelman††

*Princeton University
†University of California, Berkeley
‡International Computer Science Institute
§Consumers Behavioral Insights Lab, Bar-Ilan University, Israel
amathur@cs.princeton.edu nmalkin@cs.berkeley.edu mharbach@icsi.berkeley.edu
eyal.peer@biu.ac.il egelman@cs.berkeley.edu

What about sys admins?
Research Questions

Update Processes

What are the update steps, processes, and workflows of admins?

Impact On Effectiveness

What are the consequences of admin actions / decisions?
Study Method

Multi-part user study
1. Pilot interviews (n=7)
2. Online Survey (n=102)
3. Semi-structured interviews (n=17)

Recruitment: social media, email lists, Reddit, LISA

Screening: 18+ yrs old, US residents, employed as admin for 1+ yr at org with 5+ employees

Analysis Approach: inductive thematic analysis
Update Process Stages

Stage 1: Learning about updates
Stage 2: Deciding to update
Stage 3: Preparing for installation
Stage 4: Deploying updates
Stage 5: Handling post-update issues
Update Process Stages

Stage 1: Learning about updates
Stage 2: Deciding to update
Stage 3: Preparing for installation
Stage 4: Deploying updates
Stage 5: Handling post-update issues

- Proactive search
- Update info spread out
Update Process Stages

- Stage 1: Learning about updates
- Stage 2: Deciding to update
- Stage 3: Preparing for installation
- Stage 4: Deploying updates
- Stage 5: Handling post-update issues

- Prioritize by update type
- Proactive search
- Update info spread out
Update Process Stages

Stage 1: Learning about updates
Stage 2: Deciding to update
Stage 3: Preparing for installation
Stage 4: Deploying updates
Stage 5: Handling post-update issues

- Proactive search
- Prioritize by update type
- Update info spread out
3. Preparing For Installation

- Test Updates
- Make Backups or Snapshots
- Prepare Host Setup or Configs
3. Preparing For Installation

- **Test Updates**
  - Make Backups or Snapshots
  - Prepare Host Setup or Configs

- **Prepare Host**
  - Setup or Configs

- **No Testing** ($\approx 10\%$ in survey)

- **Dedicated Test Environment**
  - (1/3 in survey, 1/2 in interview)

- **Staggered Deployment**
  - (1/2 in survey, 2/3 in interview)

---

- **Admin Machines**
- **Developer Machines**
- **Production Machines**
3. Preparing For Installation

- No Testing ($\approx 10\%$ in survey)
- Dedicated Test Environment (1/3 in survey, 1/2 in interview)
- Staggered Deployment (1/2 in survey, 2/3 in interview)
3. Preparing For Installation

- No Testing ($\approx 10\%$ in survey)
- Dedicated Test Environment (1/3 in survey, 1/2 in interview)
- Staggered Deployment (1/2 in survey, 2/3 in interview)
3. Preparing For Installation

- No Testing ($\approx 10\%$ in survey)
- Dedicated Test Environment (1/3 in survey, 1/2 in interview)
- Staggered Deployment (1/2 in survey, 2/3 in interview)
3. Preparing For Installation

- No Testing ($\approx 10\%$ in survey)
- Dedicated Test Environment (1/3 in survey, 1/2 in interview)
- Staggered Deployment (1/2 in survey, 2/3 in interview)

Patch dev/test servers first. Let the systems run for a few weeks before patching production.

Install on non-important machines and let them bake for 1+ months.
Update Process Stages

1. **Stage 1** Learning about updates
   - Proactive search
   - Update info spread out

2. **Stage 2** Deciding to update
   - Prioritize by update type

3. **Stage 3** Preparing for installation
   - Widespread testing
   - Testing can delay update

4. **Stage 4** Deploying updates

5. **Stage 5** Handling post-update issues
Update Process Stages

Stage 1: Learning about updates
- Proactive search
- Update info spread out

Stage 2: Deciding to update
- Prioritize by update type

Stage 3: Preparing for installation
- Widespread testing
- Testing can delay update

Stage 4: Deploying updates

Stage 5: Handling post-update issues
4. Deploying Updates

Top Deployment Methods

- **Survey**
- **Interview**

<table>
<thead>
<tr>
<th>% Respondents</th>
<th>3rd-Party Update Managers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Survey: 50%</td>
</tr>
<tr>
<td></td>
<td>Interview: 75%</td>
</tr>
</tbody>
</table>

*Top Deployment Methods: Ansible, Chef*
4. Deploying Updates

Top Deployment Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Survey</th>
<th>Interview</th>
</tr>
</thead>
<tbody>
<tr>
<td>3rd-Party Update Managers</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>Custom Scripts/Programs</td>
<td>50</td>
<td>25</td>
</tr>
</tbody>
</table>
4. Deploying Updates

- **Top Deployment Methods**
  - 3rd-Party Update Managers
  - Custom Scripts/Programs
  - Automatic Updates

Survey vs. Interview

- % Respondents
  - 3rd-Party Update Managers: Survey 60%, Interview 70%
  - Custom Scripts/Programs: Survey 50%, Interview 40%
  - Automatic Updates: Survey 40%, Interview 30%
4. Deploying Updates

Survey
Interview

% Respondents

Top Deployment Methods

- 3rd-Party Update Managers
- Custom Scripts/Programs
- Automatic Updates
- Manual Updates
4. Deploying Updates

Challenges in Automation:
- High implementation cost
- Managing configs/dependencies
- Host heterogeneity
- Disruptions (more on timing in paper)

Top Deployment Methods:
- 3rd-Party Update Managers
- Custom Scripts/Programs
- Automatic Updates
- Manual Updates
Update Process Stages

Stage 1: Learning about updates
- Proactive search
- Update info spread out

Stage 2: Deciding to update
- Prioritize by update type

Stage 3: Preparing for installation
- Widespread testing
- Testing can delay update

Stage 4: Deploying updates
- Deploying + automation hard

Stage 5: Handling post-update issues
Update Process Stages

Stage 1: Learning about updates
- Proactive search
- Update info spread out

Stage 2: Deciding to update
- Prioritize by update type

Stage 3: Preparing for installation
- Widespread testing
- Testing can delay update

Stage 4: Deploying updates
- Deploying + automation hard

Stage 5: Handling post-update issues
5. Handling Post-Update Issues

98% in survey have dealt with buggy updates.

_I stopped applying updates because it was becoming more of a problem to apply them than not to._

_Production machines, they don’t get updates._
5. Handling Post-Update Issues

98% in survey have dealt with buggy updates. I stopped applying updates because it was becoming more of a problem to apply them than not to. Production machines, they don’t get updates.

Most rollback to snapshot or revert update. 5-15% try to work around update issues.
Update Process Stages

Stage 1: Learning about updates
- Proactive search
- Update info spread out

Stage 2: Deciding to update
- Prioritize by update type

Stage 3: Preparing for installation
- Widespread testing
- Testing can delay update

Stage 4: Deploying updates
- Deploying + automation hard

Stage 5: Handling post-update issues
- Most undo update, vulnerable
Update Process Stages

Stage 1: Learning about updates
Stage 2: Deciding to update
Stage 3: Preparing for installation
Stage 4: Deploying updates
Stage 5: Handling post-update issues

Organizational / Management Influences:
- Autonomy
- Policies / Compliance
- Coordination
- Resources / Budget
Where Next?

Stage 1: Learning about updates
Stage 2: Deciding to update
Stage 3: Preparing for installation
Stage 4: Deploying updates
Stage 5: Handling post-update issues

Organizational / Management Influences:
- Proactive search
- Prioritize by update type
- Update info spread out
- Widespread testing
- Testing can delay update
- Deploying + automation hard
- Most undo update, vulnerable
Where Next?

Stage 1
Learning about updates

Stage 2
Deciding to update

Stage 3
Preparing for installation

Stage 4
Deploying updates

Stage 5
Handling post-update issues

Improve information retrieval:
- Centralized source
- Outreach campaigns

Organizational / Management Influences
- Proactive search
- Update info spread out
- Prioritize by update type
- Widespread testing
- Deploying + automation hard
- Most undo update, vulnerable
- Testing can delay update
Where Next?

- **Stage 1**: Learning about updates
- **Stage 2**: Deciding to update
- **Stage 3**: Preparing for installation
- **Stage 4**: Deploying updates
- **Stage 5**: Handling post-update issues

Organizational / Management Influences:
- Proactive search
- Update info spread out
- Prioritize by update type
- Widespread testing
- Testing can delay update
- Deploying + automation hard
- Most undo update, vulnerable

- **Where Next?**
  - Standardize available info
  - Split updates by type?
Where Next?

Advance technical + usability aspects of update tools/systems

Stage 3
Preparing for installation

Stage 4
Deploying updates

Stage 5
Handling post-update issues

Organizational / Management Influences

• Proactive search
• Update info spread out

• Prioritize by update type

• Widespread testing
• Testing can delay update

• Deploying + automation hard

• Most undo update, vulnerable

Improve technical & usability aspects of update tools/systems

Advance technical + usability aspects of update tools/systems
Where Next?

Evaluate organizational influences in depth

Stage 1
Learning about updates

Stage 2
Deciding to update

Stage 3
Preparing for installation

Stage 4
Deploying updates

Stage 5
Handling post-update issues

Organizational / Management Influences

- Proactive search
- Update info spread out
- Prioritize by update type
- Widespread testing
- Deploying + automation hard
- Testing can delay update
- Most undo update, vulnerable
Where Next?

Further investigate sys admins specifically

Make information retrieval + processing easier:
- Centralized source
- Outreach campaigns
- Standardize available info
- Split update types (?)

Advance technical + usability aspects of update technology

Evaluate org influences

Thanks!

frankli@berkeley.edu
@frankli714