NessCat

By David Mberingabo

What is NessCat?

- NessCat is a tool that categorizes vulnerabilities according to a template and summarizes Nessus scan results into a report.
- NessCat contains two Python scripts:
  - Template.py: maintains the Template used during vulnerability categorization.
  - Categorizer.py: summarizes Nessus scan results following a report format.

- Types of users:
  - ***Template Keepers*** use the Template.py script
  - ***Categorizers*** use the Categorizer.py script

What is a NessCat Template

- A CSV file that defines the categorization of vulnerabilities.
- Columns in the Template:
  - ‘*Name*’ represents vulnerability.
  - ‘*Category*’ is the category given to a name.
- The Template is stored online and downloaded each time users run the Categorizer.py script.
- The Template is maintained by authorized (GitHub username and token) Template Keepers.

Who is a Template Keeper?

- Template Keepers have write access to the online Template.

- Their mission is to maintain the Template by accepting *Proposed Changes* and adding *Archived Reports* to the Template.

- They must come to consensus on the categorization of each vulnerability name.

What is a *Historical Report*?

- CSV file that contains rows of Nessus vulnerabilities that have already been categorized in the past.
- Used to initialize the Template.

How To Use NessCat as a Template Keeper?

- Step 1: Start with an empty Template.
- Step 2: Add all the *Historical Reports* to the Template, at once and go through the process while referencing the Activity Diagram.

- Example Command: `$ python3 Template.py "TestCSVs/Historical Nessus Report.csv" "TestCSVs/Historical Nessus Report_1.csv" "TestCSVs/Historical Nessus Report_2.csv"`

What is a Categorizer?

- Read access only to the online Template.
- Can automatically summarize new Nessus scan results using Categorizer.py script.
- Can create *Proposed Changes* while running Categorizer.py script.

What are *Proposed Changes*?

- CSV files that are generated if the offline template was modified, during the Categorizer.py script.
- These proposed changes should then be sent to Template Keepers for them to either add them to the Template or reject them.

How to use NessCat as a Categorizer?

- Step 1: Categorize and summarize 'Raw Nessus Scan_3.csv' and go through the process while referencing the Activity Diagram.

Technical Requirements

- Linux OS Terminal
- Python3
- Wget
- Git (for Template Keepers only)
- Python libraries (tabulate, pandas, etc…)

Further References

- https://github.com/drakstik/NessCat
- Activity Flow Diagram: https://www.figma.com/file/TaR8BULRDG4DPtaF29UAwB/Activity-Flow-Diagrams?node-id=0%3A1