Bring back RSS for operational security

Alexandre Dulaunoy
@adulau@infosec.exchange
@a@paperbay.org

TLP:CLEAR

CIRCL
Computer Incident
Response Center
Luxembourg

History of RSS

- RSS 0.90 (Really Simple Syndication) was **originally released on March 15, 1999**, by Netscape.
- It evolved into a simpler format with RSS 0.91, where the RDF elements were removed, developed by Dave Winer.
- **RSS 1.0** was later developed by the RSS-DEV Working Group, followed by the release of RSS 2.0.
- The IETF later published the **Atom Syndication Format** (RFC 4287), a more consistent format[1], yet it remains closely related to the RSS format.

[1] It's worth noting that RSS 1.0 also normalized the use of RDF, enabling compatibility with Activity Streams and other data-sharing standards.

Format and Concept

- Atom and RSS are **XML-based[2] documents used to describe feeds**, which consist of lists of related information.
- Each feed comprises a list of items or entries, each described with metadata such as title, publication date, and link.
- These formats are commonly used for updates of web content and audio content, like podcasts.
- However, the concept also **extends to information security**, where lists of updated items are crucial for tracking security events, updating threat intelligence feeds, and tracking log files.

[2] https://www.jsonfeed.org/

Security Operation Center and RSS

- Security Analysts have a **broad range of responsibilities**, including collecting events, detecting threats, analyzing them, and responding accordingly.
- **Automation is crucial**, even vital, to prevent **analyst fatigue** by streamlining repetitive tasks.
- RSS can facilitate various aspects of security operations, such as triage, vulnerability management, proactive threat monitoring, and workflow execution.

Vulnerability Management

- Vulnerability sources have diversified; the **NVD CVE is no longer the sole source of vulnerability information** due to the emergence of CNA and vendor feeds like CSAF.
- We have developed the open-source software, vulnerability-Lookup[3], which manages a multi-source vulnerability database.
- Features include an OpenAPI and an RSS feed[4] for each source of vulnerability.

[3] `https://vulnerability.circl.lu/` - `https://github.com/cve-search/vulnerability-lookup`

[4] `https://vulnerability.circl.lu/recent/nvd.atom` `https://vulnerability.circl.lu/recent/csaf_siemens.rss`

Gathering Threat Intelligence

- **All Mastodon accounts have an RSS feed available by default**. Simply take any Mastodon handle and append `.rss` to it.
- For example, on social.circl.lu, there are two bots[5]:
  - A bot from `ransomlook.io`[6] pushes a toot for each new victim.
  - A NoName57 bot[7] pushes configurations of their DDoS client as a toot.
- It's trivial to create a quick script or even use an RSS reader to filter for **information pertinent to your own organization or interests**.

[5] These bots are automatic and machine-generated. We do not control these bots.

[6] `https://social.circl.lu/@Ransomlook.rss`

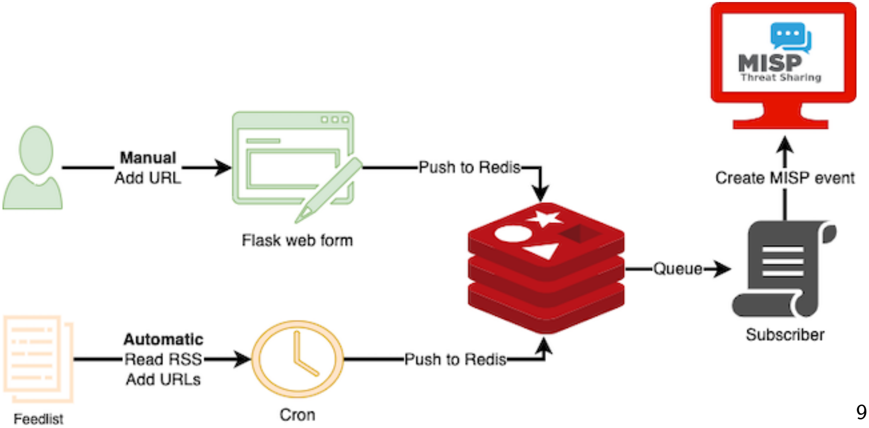
[7] `https://social.circl.lu/@NoName57Bot.rss`

Dashboarding SOC Activities

- **Standard self-hosted RSS readers** can be used to aggregate events delivered by RSS.
- Newspipe[8] is an open-source, self-hosted solution written in Python, ideal for such applications.
- This **multi-user solution** can accommodate a SOC with multiple analysts, enhancing collaborative efforts.
- It supports OPML for easy export and import of feed data, facilitating **data sharing and management**.

[8] https://github.com/cedricbonhomme/newspipe

Generating Threat Intelligence from RSS sources

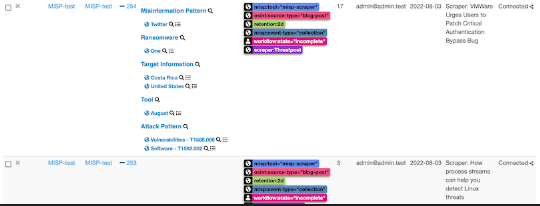


[9]

[9] https://www.misp-project.org/2022/08/08/MISP-scraper.html/

Generating Threat Intelligence from RSS sources

- **Parses RSS feeds**;
- Extracts the URLs from these feeds;
- Creates a MISP event for each URL. If the combination “event-URL” already exists then the event creation is skipped;
- Adds a MISP report (with the content of the URL) to the MISP event;
- And then uses the **report feature to extract indicators and context from the web page**;
- It is also possible to manually add URLs and outdated events are automatically deleted.



□	×	MISP-test	MISP-test	— 254	Misinformation Pattern Twitter Ransomware One Target Information Costa Rica United States Tool August Attack Pattern Vulnerabilities - T1588.006 Software - T1592.002	misp:tool="misp-scraper" osint:source-type="blog-post" retention:2d misp:event-type="collection" workflow:state="incomplete" scraper:Threatpost	17	admin@admin.test	2022-08-03	Scraper: VMWare Urges Users to Patch Critical Authentication Bypass Bug	Connected
□	×	MISP-test	MISP-test	— 253		misp:tool="misp-scraper" osint:source-type="blog-post" retention:2d misp:event-type="collection" workflow:state="incomplete"	3	admin@admin.test	2022-08-03	Scraper: How process streams can help you detect Linux threats	Connected

Combining RSS Feeds

- Sometimes you have **too many feeds or producers to process**, and you may want to merge these feeds.
- `rssmerge.py`[10] is a simple script designed to aggregate RSS feeds and merge them in reverse chronological order.
- It outputs the merged content in text, HTML, or Markdown format.
- This tool is useful for tracking recent events from various feeds, publishing them on your website, or integrating them into your SOC pipelines.

[10] https://github.com/adulau/rss-tools/blob/master/bin/rssmerge.py

Discovering RSS Feeds

- If you possess a large set of URLs from various sources and need to determine if they include an RSS feed, `rssfind.py` can help.
- `rssfind.py`[11] is designed to automate the discovery of RSS feeds.
- The script employs two techniques to identify RSS or Atom feeds:
  - The first method searches for direct link references to feeds within the HTML of a page.
  - The second method adopts a brute-force approach, trying a series of known paths to see if they lead to valid RSS or Atom feeds.

[11] `https://github.com/adulau/rss-tools/blob/master/bin/rssfind.py`

Conclusion and Future Work

- We aim to enhance the vulnerability-lookup tool with extended functionalities for RSS and Atom feeds.
- Improved integration of MISP with RSS, including support for exporting MISP data in the default format via RSS.
- Development of a method to convert MISP feeds into RSS format.
- Emphasize that **RSS is not dead; it is very much alive** and continues to be relevant in various applications.

References

- `https://github.com/adulau/rss-tools` - A set of crappy Python scripts to handle RSS in an Unix way.
- `https://www.misp-project.org/2022/08/08/MISP-scraper.html/` and `https://github.com/cudeso/misp-scraper` - MISP web scraper.