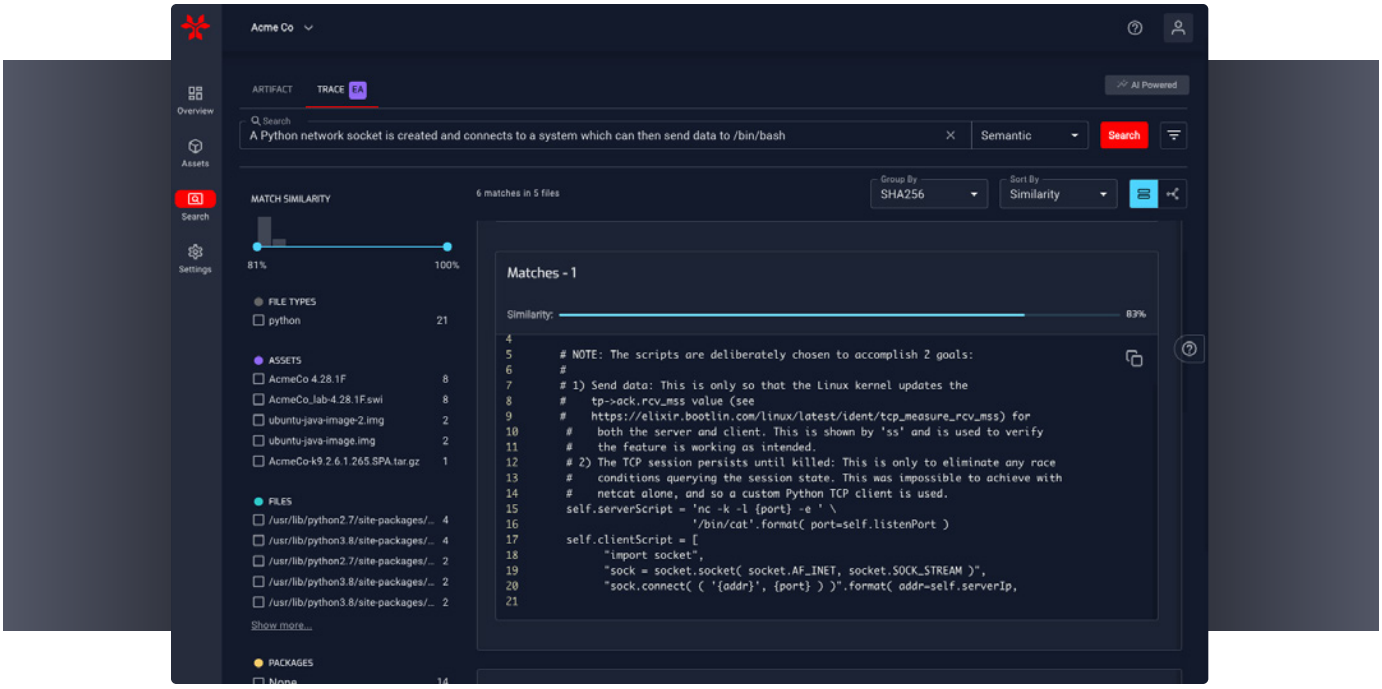
NETRISE

AI-Powered Semantic Intent Search

For The NetRise Platform

NetRise Trace is a solution at the cutting edge of semantic search capabilities, utilizing AI-powered intent interpretation to allow users to look for threats, weakness, and vulnerabilities within their software/firmware supply chain. This can quickly reveal the motives behind code and configurations rather than relying on traditional methods. Trace helps organizations quickly find impacted assets with a single query, creating a comprehensive graph of affected software supply chain components and their associated vulnerabilities. This eliminates the need for repetitive scans and accelerates the response to threats across devices, firmware, and software packages.



Trace search results are returned as blocks of text within each unique file so users can easily see the affected parts of the file even if that file exists in different locations on multiple assets.

Trace is the first solution to integrate AI-driven semantic search, supply chain impact analysis, and vulnerability validation by utilizing large language model (LLM) capabilities. The resulting solution offers customers a unified and potent method for detecting known and hidden threats in low-level firmware and other cyber-physical systems.

netrise.io | sales@netrise.io

Key Benefits of NetRise Trace

01 | AI-Powered Search

Semantic and keyword-based search for files, configurations, and vulnerabilities using AI.

A Python network socket is created and connects to a system which can then send data to /bin/bash

02 | Deep Supply Chain Introspection & Origin Tracing

Discover and trace the origin of code and risk to third-party or proprietary software packages.

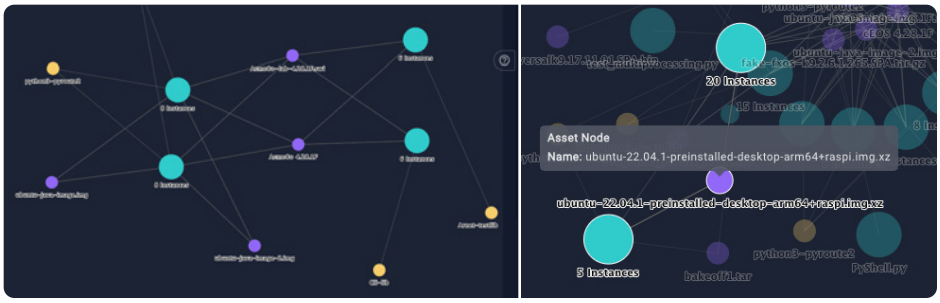
```
5       # NOTE: The scripts are deliberately chosen to accomplish 2 goals:
6       #
7       # 1) Send data: This is only so that the Linux kernel updates the
8       #    tp->ack.rcv_mss value (see
9       #    https://elixir.bootlin.com/linux/latest/ident/tcp_measure_rcv_mss) for
10       #    both the server and client. This is shown by 'ss' and is used to verify
11       #    the feature is working as intended.
12      # 2) The TCP session persists until killed: This is only to eliminate any race
13      #    conditions querying the session state. This was impossible to achieve with
14      #    netcat alone, and so a custom Python TCP client is used.
15      self.serverScript = 'nc -k -l {port} -e ' \
16                          '/bin/cat'.format( port=self.listenPort )
17      self.clientScript = [
18            "import socket",
19            "sock = socket.socket( socket.AF_INET, socket.SOCK_STREAM )",
20            "sock.connect( ( '{addr}', {port} ) )".format( addr=self.serverIp,
21
```

03 | LLM-Based Vulnerability Discovery & Validation

Identify vulnerabilities and assess their impact across an organization's firmware, software, and cyber-physical systems.

04 | Visualize Supply Chain Risks

Easily view the relationship between vulnerable code, files, packages and assets with the NetRise Supply Chain Graph View.



How Trace Works

Trace uses Text Embedding technology to create a representation of humans' natural language that a computer can understand and operate on.

Essentially, text embeddings allow computers to understand the "intent" of a particular script, block of code, or text file. The Natural Language Processing (NLP) capability of Trace allows a human to ask questions in ways we understand, which Trace can quickly and intelligently interpret to seek out similar results in either meaning or intent.

Trace search results are returned as blocks of text within each unique file so users can easily see the affected parts of the file even if that file exists in different locations on multiple assets.

Ready for a Demo?

netrise.io | sales@netrise.io