

BEAM Virtual Machine shenanigans

TLP:CLEAR



CIRCL
Computer Incident
Response Center
Luxembourg

Jean-Louis Huynen¹

¹CIRCL

23/10/2024

Preleminaries

- Erlang/Elixir source code is compiled into `.beam` bycode
- BEAM is the virtual machine that runs this bycode
- The BEAM runs as a system process, and holds Erlang processes
- Processes inside the BEAM VM exchanges messages through mail boxes
- The Open Telecom Platform contains:
 - the BEAM,
 - the erlang compiler,
 - behaviours and applications for developing horizontally scalable systems process groups

Preliminaries

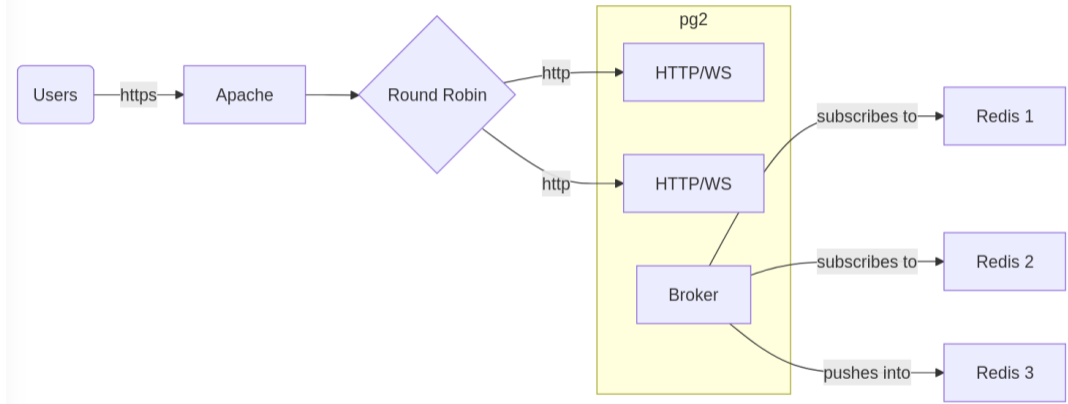


Figure 1: use of pg2 in cocktailparty

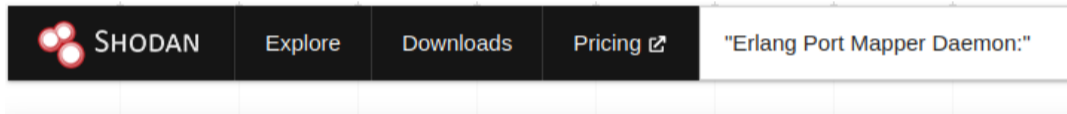
Distributed Erlang

- Erlang runtimes (nodes) communicating with each other
- Protected with a PSK: the cookie
- Erlang Port Mapper Daemon maps nodes names to system's port

```
$ echo -n -e "\x00\x01\x6e" | nc -vn 10.56.244.246 4369
Connection to 10.56.244.246 4369 port [tcp/*] succeeded!
name erlol at port 44101
```

Distributed Erlang

- EPMD should not be exposed to the internet
- node communications should never occur over untrusted networks



The screenshot shows the SHODAN search interface. The top navigation bar includes the SHODAN logo, 'Explore', 'Downloads', and 'Pricing' with an external link icon. The search query is '"Erlang Port Mapper Daemon:"'. Below the navigation bar, the search results are displayed. The 'TOTAL RESULTS' section shows 82,746 results. The 'TOP COUNTRIES' section shows a map of China with a red overlay, indicating that the majority of results are from China. On the right side, there are options to 'View' results and a 'Part' button.

TOTAL RESULTS

82,746

TOP COUNTRIES

 View

Part

211.14

Chengdu
LTD

 China

Eshell

- Connects your local node to a remote elixir or erlang REPL running on BEAM
- Very bad logging by default
- can compile from String
 - `[{module, binary}] = Code.compile_string("defmodule CompTest do def print(x) do IO.puts(x) end end")`
- can replace an existing module
- without the source: get the AST, modify the AST, recompile (left as an exercise :-)

Eshell

- can deploy on all cluster's nodes.
 - erlang nl(module)
 - elixir
 - Enum.each(Node.list(), fn node -> :rpc.call(node, :code, :load_binary, [module, 'nofile', binary]) end)
 - Enum.each(Node.list(), fn node -> :rpc.call(node, CompTest, :print, ["Hello from #{node}!"]) end)

Deploy new modules

- reverse shell on the host
- ssh server for encrypted communication
- ssh client for pivoting and scanning
- defensive module
- special sauce :-)

Conclusion

- Look into the <https://erlef.org/wg/security> before deploying
- Think twice before connecting to untrusted nodes

Credits and References

- <https://github.com/flowintel/cocktailparty>
- <https://www.erlang.org/docs/18/man/pg2.html>
- <https://book.hacktricks.xyz/network-services-pentesting/4369-pentesting-erlang-port-mapper-daemon-epmd>
- <https://github.com/gteissier/erl-matter>
- <https://www.broot.ca/erlang-remsh-is-dangerous.html>