

mtrudel / bandit Public

<> Code Issues 1 Pull requests 1 Actions Security and quality 5 Ins

# Commit 45feea2

mtrudel authored 11 hours ago · 23 / 27 · Verified

Merge commit from fork

main · 1.11.0

1 parent [f2ca636](#) commit 45feea2

3 files changed

+10 -15

↑ Top

Filter files...

- lib/bandit
  - pipeline.ex
- test/bandit
  - http1
    - protocol\_test.exs
  - http2
    - protocol\_test.exs

Search within code

lib/bandit/pipeline.ex

```

@@ -73,22 +73,17 @@ defmodule Bandit.Pipeline do
73 73         ) :: Plug.Conn.t()
74 74         defp build_conn!(transport, method, request_target, headers, {secure?,
           peer_address}, opts) do
75 75         adapter = Bandit.Adapter.init(self(), transport, method, headers, opts)

```

```

76 -   scheme = determine_scheme(secure?, request_target)
76 +   scheme = determine_scheme(secure?)
77 77   version = Bandit.HTTPTransport.version(transport)
78 78   {host, port} = determine_host_and_port!(scheme, version, request_target,
      headers)
79 79   {path, query} = determine_path_and_query(request_target)
80 80   uri = %URI{scheme: scheme, host: host, port: port, path: path, query: query}
81 81   Plug.Conn.Adapter.conn({Bandit.Adapter, adapter}, method, uri, peer_address,
      headers)
82 82   end
83 83
84 -   @spec determine_scheme(boolean(), request_target()) :: String.t() | nil
85 -   defp determine_scheme(secure?, {scheme, _, _, _}) do
86 -     case {secure?, scheme} do
87 -     {true, nil} -> "https"
88 -     {false, nil} -> "http"
89 -     {_, scheme} -> scheme
90 -     end
91 -   end
84 +   @spec determine_scheme(boolean()) :: String.t()
85 +   defp determine_scheme(true), do: "https"
86 +   defp determine_scheme(false), do: "http"
92 87
93 88   @spec determine_host_and_port!(binary(), atom(), request_target(),
      Plug.Conn.headers()) ::
94 89     {Plug.Conn.host(), Plug.Conn.port_number()}

```

```

▼ test/bandit/http1/protocol_test.exs ...
↑... @@ -520,11 +520,11 @@ defmodule HTTP1ProtocolTest do
520 520   end
521 521
522 522   describe "absolute-form request target (RFC9112§3.2.2)" do
523 -     test "uses request-line scheme even if it does not match the transport",
      context do
523 +     test "uses transport scheme even if it does not match request-line scheme",
      context do
524 524       client = SimpleHTTP1Client.tcp_client(context)
525 525       SimpleHTTP1Client.send(client, "GET", "https://banana/echo_components")

```

```

526 526         assert {:ok, "200 OK", _headers, body} =
              SimpleHTTP1Client.recv_reply(client)
527 -         assert Jason.decode!(body)["scheme"] == "https"
527 +         assert Jason.decode!(body)["scheme"] == "http"
528 528         end
529 529
530 530         test "derives host from the URI, even if it differs from host header",
              context do

```

test/bandit/http2/protocol\_test.exs

```

@@ -2649,7 +2649,7 @@ defmodule HTTP2ProtocolTest do
2649 2649         assert Jason.decode!(body)["scheme"] == "https"
2650 2650         end
2651 2651
2652 -         test "uses :scheme even if it does not match transport", context do
2652 +         test "uses transport even if it does not match :scheme", context do
2653 2653             socket = SimpleH2Client.setup_connection(context)
2654 2654
2655 2655             headers = [
@@ -2663,7 +2663,7 @@ defmodule HTTP2ProtocolTest do
2663 2663
2664 2664         assert SimpleH2Client.successful_response?(socket, 1, false)
2665 2665         {:ok, 1, true, body} = SimpleH2Client.recv_body(socket)
2666 -         assert Jason.decode!(body)["scheme"] == "http"
2666 +         assert Jason.decode!(body)["scheme"] == "https"
2667 2667         end
2668 2668
2669 2669         test "derives host from host header", context do
@@ -2940,7 +2940,7 @@ defmodule HTTP2ProtocolTest do
2940 2940         assert Jason.decode!(body)["scheme"] == "https"
2941 2941         end
2942 2942
2943 -         test "uses :scheme even if it does not match transport", context do
2943 +         test "uses transport even if it does not match :scheme", context do
2944 2944             socket = SimpleH2Client.setup_connection(context)
2945 2945
2946 2946             headers = [
@@ -2954,7 +2954,7 @@ defmodule HTTP2ProtocolTest do

```

```
2954 2954
2955 2955     assert SimpleH2Client.successful_response?(socket, 1, false)
2956 2956     {:ok, 1, true, body} = SimpleH2Client.recv_body(socket)
2957 2957     - assert Jason.decode!(body)["scheme"] == "http"
2957 2957     + assert Jason.decode!(body)["scheme"] == "https"
2958 2958     end
2959 2959
2960 2960     test "derives host from :authority header, even if it differs from host
header", context do
```

## Comments 0



Please [sign in](#) to comment.