Commit 8fc71ac

Whaileee committed 3 weeks ago Verified

```
inets: Check script_alias when using mod_auth
```

master · OTP-29.0-rc3 ••• OTP-27.3.4.10

1 parent c388a2d commit 8fc71ac

3 files changed +94 -4 lines changed

Top

Filter files…

- lib/inets
  - src/http_server
    - httpd.erl
    - mod_alias.erl
  - test
    - httpd_SUITE.erl

3 files changed +94 -4 lines changed

Search within code

lib/inets/src/http_server/httpd.erl

```
@@ -435,6 +435,14 @@ property list.
435 435   Access to http://your.server.org/cgi-bin/foo would cause the server to run
      the
436 436   script /web/cgi-bin/foo.
437 437
    438 +   > #### Note {: .info }
    439 +   >
    440 +   > When using `script_alias` with directory-based authentication
```

```
      441  +   > (see [`directory`](`m:httpd#prop_dri`)), ensure that authentication
      442  +   > rules reference the actual filesystem path (RealName), not the URL path
           (Alias).
      443  +   > The server correctly resolves script_alias paths for authentication checks.
      444  +   >
      445  +
438   446     - [](){: #prop_script_re_write } **`{script_re_write, {Re, Replacement}}`**
439   447       `Re = string()` and `Replacement = string()`. Have the same behavior as
440   448       property `re_write`, except that they also mark the target directory as
```

lib/inets/src/http_server/mod_alias.erl

```
           @@ -309,6 +309,13 @@ store({re_write, {Re, Replacement}} = Conf, _)
309   309         end;
310   310     store({re_write, _} = Conf, _) ->
311   311         {error, {wrong_type, Conf}};
      312  +
      313  + % When `script_alias` is used in conjunction with `m:mod_auth` for directory-
           based
      314  + % access control, authentication rules are evaluated against the actual
           filesystem
      315  + % path where scripts reside, not the aliased URL path. This ensures that CGI
           scripts
      316  + % mapped outside the document root are properly protected by directory
           authentication
      317  + % directives.
      318  +
312   319     store({script_alias, {Fake, Real}}, _)
313   320       when is_list(Fake), is_list(Real) ->
314   321         {ok, {script_alias,{"^"++Fake,Real}}};
           @@ -339,7 +346,8 @@ is_directory_index_list(_) ->
339   346     %% ----------------------------------------------------------------------
340   347
341   348     which_alias(ConfigDB) ->
342       -     httpd_util:multi_lookup(ConfigDB, alias).
      349  +     httpd_util:multi_lookup(ConfigDB, alias) ++
      350  +         httpd_util:multi_lookup(ConfigDB, script_alias).
343   351
344   352     which_document_root(ConfigDB) ->
```

```
345  353        Root = httpd_util:lookup(ConfigDB, document_root, ""),
```

lib/inets/test/httpd_SUITE.erl

```
@@ -79,6 +79,7 @@ all() ->
79   79         {group, http_logging},
80   80         {group, http_post},
81   81         {group, http_rel_path_script_alias},
     82 +       {group, http_script_alias_auth},
82   83         {group, http_not_sup},
83   84         {group, https_alert},
84   85         {group, https_not_sup},
@@ -142,6 +143,7 @@ groups() ->
142  143         {http_1_0, [], [{group, http_1_0_parallel} | load()]},
143  144         {http_1_0_parallel, [parallel], [host, cgi, trace] ++ http_head() ++
     http_get()},
144  145         {http_rel_path_script_alias, [], [cgi]},
     146 +       {http_script_alias_auth, [], [script_alias_auth_bypass]},
145  147         {esi, [], [erl_script_timeout_default,
146  148                    erl_script_timeout_option,
147  149                    erl_script_timeout_proplist,
@@ -275,6 +277,9 @@ init_per_group(http_logging, Config) ->
275  277    init_per_group(http_rel_path_script_alias = Group, Config) ->
276  278        ok = start_apps(Group),
277  279        init_httpd(Group, [{type, ip_comm},{http_version, "HTTP/1.1"}| Config]);
     280 + init_per_group(http_script_alias_auth = Group, Config) ->
     281 +     ok = start_apps(Group),
     282 +     init_httpd(Group, [{type, ip_comm},{http_version, "HTTP/1.1"}| Config]);
278  283    init_per_group(not_sup, Config) ->
279  284        [{http_version, "HTTP/1.1"} | Config];
280  285    init_per_group(Group, Config) when Group == esi ->
@@ -296,6 +301,7 @@ end_per_group(Group, _Config) when Group ==
     http_basic;
296  301                                       Group == http_mime_type;
297  302                           Group == http_mime_and_default_type;
298  303                                       Group == http_mime_types;
     304 +                                     Group == http_script_alias_auth;
299  305                                       Group == esi
```

```
300  306                          ->
301  307          inets:stop();
     @@ -1139,6 +1145,34 @@ cgi(Config) when is_list(Config) ->
1139 1145                 [{statuscode, 200},
1140 1146                  {no_header, "cache-control"}]).
1141 1147     %%--------------------------------------------------------------------
     1148 + script_alias_auth_bypass() ->
     1149 +     [{doc, "Test that mod_auth correctly protects script_alias directories "
     1150 +            "outside DocumentRoot (CVE-2026-28808)"}].
     1151 + script_alias_auth_bypass(Config) when is_list(Config) ->
     1152 +     Version = proplists:get_value(http_version, Config),
     1153 +     Host = proplists:get_value(host, Config),
     1154 +     Script =
     1155 +         case os:type() of
     1156 +             {win32, _} -> "printenv.bat";
     1157 +             _ -> "printenv.sh"
     1158 +         end,
     1159 +     %% Unauthenticated request must be rejected with 401
     1160 +     ok = http_status("GET /http_script_alias_auth/" ++ Script ++ " ", Config,
     1161 +                      [{statuscode, 401},
     1162 +                       {header, "WWW-Authenticate"}]),
     1163 +     %% Authenticated request must succeed
     1164 +     ok = auth_status(
     1165 +            auth_request("/http_script_alias_auth/" ++ Script, "one",
          "onePassword",
     1166 +                         Version, Host),
     1167 +            Config,
     1168 +            [{statuscode, 200}]),
     1169 +     %% Wrong password must be rejected
     1170 +     ok = auth_status(
     1171 +            auth_request("/http_script_alias_auth/" ++ Script, "one",
          "WrongPassword",
     1172 +                         Version, Host),
     1173 +            Config,
     1174 +            [{statuscode, 401}]).
     1175 + %%--------------------------------------------------------------------
1142 1176     cgi_chunked_encoding_test() ->
1143 1177         [{doc, "Test chunked encoding together with mod_cgi "}].
1144 1178     cgi_chunked_encoding_test(Config) when is_list(Config) ->
```

```
@@ -2053,13 +2087,15 @@ do_max_clients(Config) ->
2053 2087
2054 2088   setup_server_dirs(ServerRoot, DocRoot, DataDir) ->
2055 2089       CgiDir =  filename:join(ServerRoot, "cgi-bin"),
     2090 +     ExtCgiDir = filename:join(ServerRoot, "ext-cgi-bin"),
2056 2091       AuthDir =  filename:join(ServerRoot, "auth"),
2057 2092       PicsDir =  filename:join(ServerRoot, "icons"),
2058 2093       ConfigDir =  filename:join(ServerRoot, "config"),
2059 2094
2060 2095       ok = file:make_dir(ServerRoot),
2061 2096       ok = file:make_dir(DocRoot),
2062 2097       ok = file:make_dir(CgiDir),
     2098 +     ok = file:make_dir(ExtCgiDir),
2063 2099       ok = file:make_dir(AuthDir),
2064 2100       ok = file:make_dir(PicsDir),
2065 2101       ok = file:make_dir(ConfigDir),
@@ -2073,6 +2109,7 @@ setup_server_dirs(ServerRoot, DocRoot, DataDir) ->
2073 2109       inets_test_lib:copy_dirs(DocSrc, DocRoot),
2074 2110       inets_test_lib:copy_dirs(AuthSrc, AuthDir),
2075 2111       inets_test_lib:copy_dirs(CgiSrc, CgiDir),
     2112 +     inets_test_lib:copy_dirs(CgiSrc, ExtCgiDir),
2076 2113       inets_test_lib:copy_dirs(PicsSrc, PicsDir),
2077 2114       inets_test_lib:copy_dirs(ConfigSrc, ConfigDir),
2078 2115
@@ -2091,7 +2128,13 @@ setup_server_dirs(ServerRoot, DocRoot, DataDir) ->
2091 2128       EnvCGI =  filename:join([ServerRoot, "cgi-bin", "printenv.sh"]),
2092 2129       {ok, FileInfo1} = file:read_file_info(EnvCGI),
2093 2130       ok = file:write_file_info(EnvCGI,
2094      -                 FileInfo1#file_info{mode = 8#00755}).
     2131 +                 FileInfo1#file_info{mode = 8#00755}),
     2132 +
     2133 +     %% Set permissions for ext-cgi-bin scripts (outside DocumentRoot)
     2134 +     ExtEnvCGI = filename:join([ServerRoot, "ext-cgi-bin", "printenv.sh"]),
     2135 +     {ok, FileInfo2} = file:read_file_info(ExtEnvCGI),
     2136 +     ok = file:write_file_info(ExtEnvCGI,
     2137 +                               FileInfo2#file_info{mode = 8#00755}).
2095 2138
2096 2139   setup_tmp_dir(PrivDir) ->
2097 2140       TmpDir =  filename:join(PrivDir, "tmp"),
```

```
@@ -2130,6 +2173,7 @@ start_apps(Group) when Group == http_basic;
2130 2173                          Group == http_mime_and_default_type;
2131 2174                          Group == http_mime_types;
2132 2175                          Group == http_rel_path_script_alias;
     2176 +                        Group == http_script_alias_auth;
2133 2177                          Group == http_not_sup;
2134 2178                          Group == http_mime_types;
2135 2179                          Group == esi ->
@@ -2249,6 +2293,20 @@ server_config(http_erl_script_alias_all, Config) -
>
2249 2293  server_config(http_rel_path_script_alias, Config) ->
2250 2294      ServerRoot = proplists:get_value(server_root, Config),
2251 2295      config_template(Config, ServerRoot, "./cgi-bin/", [httpd_example, io]);
     2296 + server_config(http_script_alias_auth, Config) ->
     2297 +     ServerRoot = proplists:get_value(server_root, Config),
     2298 +     %% CGI dir is outside DocumentRoot (sibling under ServerRoot)
     2299 +     ExtCgiDir = filename:join(ServerRoot, "ext-cgi-bin") ++ "/",
     2300 +     [{modules, [mod_alias, mod_auth, ?MODULE, mod_get, mod_head]},
     2301 +      {logger, [{error, httpd_test}]},
     2302 +      {script_alias, {"/http_script_alias_auth/", ExtCgiDir}},
     2303 +      {directory, {filename:join(ServerRoot, "ext-cgi-bin"),
     2304 +                   [{auth_type, plain},
     2305 +                    {auth_name, "Protected CGI"},
     2306 +                    {auth_user_file, filename:join(ServerRoot,
"auth/passwd")},
     2307 +                    {auth_group_file, filename:join(ServerRoot,
"auth/group")},
     2308 +                    {require_user, ["one", "Aladdin"]}]}}
     2309 +     ] ++ server_config(http, Config);
2252 2310  server_config(https, Config) ->
2253 2311      SSLConf = proplists:get_value(ssl_conf, Config),
2254 2312      ServerConf = proplists:get_value(server_config, SSLConf),
@@ -2336,9 +2394,25 @@ do(ModData) ->
2336 2394                  ok;
2337 2395              _ ->
2338 2396                  {already_sent, Status, _Size} = proplists:get_value(response,
ModData#mod.data),
2339      -                 propagate_test ! {status, Status}
```

```
                2397 +              propagate_test ! {status, Status}
2340            2398         end,
2341                -       {proceed, ModData#mod.data}.
                2399 +       case ModData#mod.request_uri of
                2400 +           "/http_script_alias_auth/" ++ _ ->
                2401 +               case proplists:get_value(status, ModData#mod.data) of
                2402 +                   {_StatusCode, _PhraseArgs, _Reason} ->
                2403 +                       {proceed, ModData#mod.data};
                2404 +                   undefined ->
                2405 +                       case proplists:get_value(response, ModData#mod.data) of
                2406 +                           undefined ->
                2407 +                               Body = "<html>script_alias_auth_bypass test
                       ok</html>",
                2408 +                               {proceed, [{response, {200, Body}} |
                       ModData#mod.data]};
                2409 +                           _Response ->
                2410 +                               {proceed, ModData#mod.data}
                2411 +                       end
                2412 +               end;
                2413 +           _ ->
                2414 +               {proceed, ModData#mod.data}
                2415 +       end.
2342            2416
2343            2417     not_sup_conf() ->
2344            2418         [{modules, [mod_get]}].
```

Comments 0

Please sign in to comment.