

Mini Tehdit İstihbaratı

written by Mert SARICA | 3 December 2018

Siber Güvenlik Merkezleri tarafından kurumlara gerçekleştirilen hedefli siber saldıruları tespit etmenin ve engellemenin öneminin yüksek olduğu günümüzde, basit bir kontrol ile tespit edilebilen düşük etkileşimli bir balküpü sistemi (Bkz: Balküpü Tespiti) maalesef kurumlar için kaynakları tüketen atıl bir sistem olmaktan öteye gidemiyor. Halbuki kolay bir şekilde kurup, yönetebileceğimiz, amacına hizmet edebilen bir balküpü sistemi yeri geldiğinde SIEM ile entegre edilerek kurumumuz için oldukça değerli bir mini siber tehdit istihbarat servisi olarak da kullanılabilir.

Bu ev yapımı, mini istihbarat servisi ile ayrıca kurumların hangi basit şifreler ile hangi servisler, protokoller üzerinden hacklendiğini, hangi ülkelerin siber saldırınrlara ev sahipliği yaptığını bulmak da mümkün olabilirdi. Tuzak Sistem ile Hacker Avı çalışmamda olduğu gibi evimde konumlandıracağım bu sistemin hem basit bir şekilde siber saldırınrlar tarafından tespit edilememesi hem de kısıtlı zamanımı çalmaması adına kolay yönetilebilmesi gereksinimlerimin başında geliyordu.

Bu konu üzerine yeteri kadar düşündükten ve taşındıktan sonra açık kaynak kodlu çeşitli servisleri modifiye edip bu servise yapılan hatalı giriş denemelerini (failed login attempt) istediğim şekilde kayıt altına alan bir sistem oluşturmaya karar verdim. Hangi servisler olacağına karar vermek için ise güvenlik uzmanları dışında bir o kadar da art niyetli kişiler tarafından da sıkılıkla kullanılan ncrack ve THC-Hydra araçlarının destekledikleri protokollere göz atmaya karar verdim ve günün sonunda ssh, ftp, http, postgresql protokollerinde karar kıldım. Daha sonra Mini-PC üzerine kurduğum XenServer sanallaştırma sistemi üzerine iki tane sanal Ubuntu işletim sistemi kurdum. Ubuntulardan birine bu denemeleri kayıt altına alan loginmon isimli aracı bir diğerine loginmon aracının kayıtlarını (log) analiz etmek amacıyla Splunk Community sürümünü kurdum.

GitHub, Inc. [US] | https://github.com/nmap/ncrack

File	Description	Last Commit
services.h	updated license header	2 months ago
sh tool	removed empty install.sh and added sh tool	9 years ago
targets.cc	updated license header	2 months ago
targets.h	updated license header	2 months ago
timing.cc	updated license header	2 months ago
timing.h	updated license header	2 months ago
utils.cc	fixed length type in base64_decode and associated warning	a month ago
utils.h	fixed length type in base64_decode and associated warning	a month ago

README.md

ncrack

Ncrack is a high-speed network authentication cracking tool. It was built to help companies secure their networks by proactively testing all their hosts and networking devices for poor passwords. Security professionals also rely on Ncrack when auditing their clients. Ncrack was designed using a modular approach, a command-line syntax similar to Nmap and a dynamic engine that can adapt its behaviour based on network feedback. It allows for rapid, yet reliable large-scale auditing of multiple hosts.

Ncrack's features include a very flexible interface granting the user full control of network operations, allowing for very sophisticated bruteforcing attacks, timing templates for ease of use, runtime interaction similar to Nmap's and many more. Protocols supported are: SSH, RDP, FTP, Telnet, HTTP(S), POP3(S), IMAP, SMB, VNC, SIP, Redis, PostgreSQL, MySQL, MSSQL, MongoDB, Cassandra, WinRM, OWA.

Be sure to read the Ncrack man page (<https://nmap.org/ncrack/man.html>) to fully understand Ncrack usage. If you are a developer and want to write your own Ncrack modules, studying the Ncrack Developer's Guide (<https://nmap.org/ncrack/devguide.html>) would be the first step.

© 2017 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub API Training Shop Blog About

THC-HYDRA - fast and ... | Not secure | https://www.thc.org/thc-hydra/



THC-Hydra

A very fast network logon cracker which support many different services.
See feature sets and services coverage [here](#) - Incl. a speed comparison against ncrack and medusa

Current Version: 8.6
Last update 2017-07-21

[0x00] News and Changelog

Check out the feature sets and services coverage [here](#) - including a speed comparison against ncrack and medusa (yes, we win :-))
Development code is available at a public github repository: <https://github.com/vanhauser-thc/thc-hydra>
There is a new section below for online tutorials.
Read below for Linux compilation notes.

CHANGELOG for 8.6

! Development moved to a public github repository: <https://github.com/vanhauser-thc/thc-hydra>

i Reports came in that the rdp module is not working reliable sometimes, most likely against new windows versions. please test, report and if possible send a fix

* added radmin module by catonic prines - great work!

* secd module now checks if SPNv1 is supported by the server and if signing is required

* http-form module now supports URLs up to 6000 bytes (thanks to petrock@github for the patch)

* fix: SQL injection attack failed with error:00000000:lib(0)::reason(0) (thanks galad@github for reporting)

* Added new command line option:
-c TIME seconds between login attempts (overwrites all threads, so -t is recommended)

* Options put after --config is reading it first, then it is ignored and loaded before

* merged several patches by Diabolo@github to make the code easier readable. thanks for that!

* merged a patch by Diabolo@github that moves the help output to the individual module

You can also take a look at the full [changes](#) file

[0x01] Introduction

Welcome to the mini website of the THC Hydra project.

Number one of the biggest security holes are passwords, as every password security study shows. Hydra is a parallelized login cracker which supports numerous protocols to attack. New modules are easy to add, beside that, it is flexible and very fast.

Hydra was tested to compile on Linux, Windows/Cygwin, Solaris 11, FreeBSD 8.1, OpenBSD, OSX, QN/Blackberry, and is made available under GPLv3 with a special OpenSSL license expansion.

Currently this tool supports:

Asterisk, AFS, Cisco AAA, Cisco enable, CVS, Firebird, FTP, HTTP-FORM-GET, HTTP-FORM-POST, HTTPS-HEAD, HTTP-GET, HTTP-HEAD, HTTP-PROXY, HTTPS-FORM-GET, HTTPS-FORM-POST, HTTPS-GET, HTTPS-POST, HTTPS-HEAD, HTTPS-PUT, IMAP, LDAP, MySQL, MySQLi, Oracle, MySQLi, PostgreSQL, PostgreSQLi, MSSQL, MSSQLi, PC-Anywhere, PCMF, POP3, POSTGRES, RDP, Reesc, Rlogin, Rsh, RTP, S7-300, SAP/SID/ABAP, SMB, SMTP, SFTP, SSH, SSH-Enum, SMTP, SOCKS5, SSH (v1 and v2), Subversion, Teampeak (T52), Telnet, VMware-Auth, VNC and XPP.

For HTTP, POP3, IMAP and SMTP, several login mechanisms like plain and MD5 digest etc. are supported.

Tabii önemdeki en büyük engellerden biri bu protokollerini kullanan sunucu bileşenlerinin (sshd, postgresql, vsftpd gibi) hatalı giriş denemelerinde kullanılan parolaların açık (clear-text) olarak kayıt altına almalarıyla bu sebeple her bir sunucu bileşeninin kaynak koduna teker teker müdahale etmeye başladım.

Öncelikle openssh-7.2p2 paketini indirip auth-passwd.c dosyasını daha sonra postgres-10.0 paketini indirip src/backend/libpq/auth.c dosyasını ve ardından da vsftpd-3.0.3 paketini indirip prelogin.c ve logging.c dosyalarını, parolaların açık halini kayıt edecek şekilde değiştirdim ve derledim.

```

Batcave X Batcave (1) Batcave (3) Batcave (2) Kali (VM)
GNU nano 2.5.3 File: auth-passwd.c

#include "canohost.h"
extern buffer logmsg;
extern Serveroptions options;
#ifndef HAVE_LOGIN_CAP
extern login_cap_t *lc;
#endif

#define DAY (24L * 60 * 60) /* 1 day in seconds */
#define TWO_WEEKS (2L * 7 * DAY) /* 2 weeks in seconds */

void disable_forwarding(void)
{
    no_port_forwarding_flag = 1;
    no_agent_forwarding_flag = 1;
    no_x11_forwarding_flag = 1;
}

/*
 * Tries to authenticate the user using password. Returns true if
 * authentication succeeds.
 */
int auth_password(Authctxt *authctxt, const char *password)
{
    struct passwd *pw = authctxt->pwd;
    int result, ok = authctxt->valid;
#if defined(USE_SHADOW) && defined(HAS_SHADOW_EXPIRE)
    static int expire_checked = 0;
#endif

    /* Mert SARICA */
    Logit("Failed Password Client: %200s Username: %s Password: %s", get_remote_ipaddr(), authctxt->user, password);
#ifndef HAVE_CYGIN
    if ((pw->uid == 0 && options.permit_root_login != PERMIT_YES)
        || (pw->uid == 0))
        ok = 0;
#endif
    if (*password == '\0' && options.permit_empty_passwd == 0)
        return 0;
#if defined(KRBS)
    if (options.kerberos_authentication == 1) {
        int ret = auth_krb5_password(authctxt, password);
        if (ret == 1 || ret == 0)
            return ret && ok;
        /* Fall back to ordinary passwd authentication. */
    }
#endif
#ifndef HAVE_CYGIN
    {
        HANDLE hToken = cygwin_logon_user(pw, password);
        if (hToken == INVALID_HANDLE_VALUE)
            return 0;
        cygwin_set_impostation_token(hToken);
        return ok;
    }
#endif
#ifndef USE_PAM
    if (options.use_pam)
        return (sshpam_auth_passwd(authctxt, password)) && ok;
#endif
#if defined(USE_SHADOW) & defined(HAS_SHADOW_EXPIRE)
    if (expire_checked) {
        ...
    }
#endif
}

```

```

root@kali:~# cat pass.txt
test
mert
dert
root@kali:~# hydra -t 32 -l root -P pass.txt 192.168.1.127 ssh
Hydra v8.3 (c) 2016 by Van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2017-11-07 11:06:04
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 3 tasks per 1 server, overall 64 tasks, 3 login tries (l:1/p:3), ~0 tries per task
[DATA] attacking service ssh on port 22
1 of 1 target completed, 0 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2017-11-07 11:06:05
root@kali:~#

```

```

root@ubuntu:~# tail -n 9 /var/log/auth.log
Nov  7 19:06:05 ubuntu sshd[20604]: [Failed Password] Client: 192.168.1.144 Username: root Password: mert
Nov  7 19:06:05 ubuntu sshd[20603]: [Failed Password] Client: 192.168.1.144 Username: root Password: dert
Nov  7 19:06:05 ubuntu sshd[20602]: [Failed Password] Client: 192.168.1.144 Username: root Password: test
Nov  7 19:06:05 ubuntu sshd[20604]: Failed password for root from 192.168.1.144 port 64781 ssh2
Nov  7 19:06:05 ubuntu sshd[20603]: Failed password for root from 192.168.1.144 port 64780 ssh2
Nov  7 19:06:05 ubuntu sshd[20602]: Failed password for root from 192.168.1.144 port 64779 ssh2
Nov  7 19:06:05 ubuntu sshd[20604]: Connection closed by 192.168.1.144 port 64781 [preauth]
Nov  7 19:06:05 ubuntu sshd[20603]: Connection closed by 192.168.1.144 port 64780 [preauth]
Nov  7 19:06:05 ubuntu sshd[20602]: Connection closed by 192.168.1.144 port 64779 [preauth]
root@ubuntu:~#

```

```
✓ Batcave X | ✓ Batcave (1) | ✓ Batcave (3) | ✓ Batcave (2) | ✓ Batcave (4)
File: auth.c
GNU nano 2.5.3

/* DO NOT echo password to logs, for security. */
elog(DEBUGS, "received password packet");

/*
 * Return the received string. Note we do not attempt to do any
 * character-set conversion on it; since we don't yet know the client's
 * encoding, there wouldn't be much point.
 */
return buf.data;
}

/*
 * Password-based authentication mechanisms
 */
/*
 * Plaintext password authentication.
 */
static int
CheckPasswordAuth(Port *port, char **logdetail)
{
    char      *passwd;        result;
    char      *shadow_pass;
    sendAuthRequest(port, AUTH_REQ_PASSWORD, NULL, 0);

    passwd = recv_Password_packet(port);
    if (passwd == NULL)
        return STATUS_EOF; /* client wouldn't send password */

    /* Mert SARICA */
    elog(LOG, "[Failed Password] Username: %s", port->user_name, passwd);
    shadow_pass = get_role_password(port->user_name, logdetail);
    if (shadow_pass)
    {
        result = plain_crypt_verify(port->user_name, shadow_pass, passwd,
                                    logdetail);
    }
    else
        result = STATUS_ERROR;

    if (shadow_pass)
        pfree(shadow_pass);
    pfree(passwd);
    return result;
}

/*
 * MD5 and SCRAM authentication.
 */
static int
CheckPwChallengeAuth(Port *port, char **logdetail)
{
    int          auth_result;
    char      *shadow_pass;
    PasswordType ptype;
    Assert(port->hba->auth_method == uasCRAM ||
           port->hba->auth_method == uamDBS);

    /* First look up the user's password. */
    shadow_pass = get_role_password(port->user_name, logdetail);
}

Get Help     Write Out     Where Is     Cut Text     Uncut Text     Justify     Cur Pos     Prev Page     Next Page     First Line     Last Line     Where Is Next     Mark Text     Indent Text     Undo
Alt-Exit     Read File     Replace     To Spell     Go To Line     To Bracket     Copy Text     Unindent Text     Ctrl-C     Ctrl-V     Ctrl-X     Ctrl-Z
15:55
```

✓ Batcave X | ✓ Batcave (1) | ✓ Batcave (3) | ✓ Batcave (2) | ✓ Batcave (4)

```
root@ubuntu:~/honeypot# hydra -l root -P pass.txt 127.0.0.1 postgres
Hydra v8.6 (c) 2017 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2017-11-05 15:56:53
[DATA] max 3 tasks per 1 server, overall 3 tasks, 3 login tries (1:1:p:3), ~1 try per task
[DATA] attacking postgres://127.0.0.1:5432/
1 of 1 target completed, 0 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2017-11-05 15:56:54
root@ubuntu:~/honeypot# cat /var/log/pgsql
2017-11-05 15:56:27.206 +03 [10974] LOG:  listening on IPv6 address "::1", port 5432
2017-11-05 15:56:27.206 +03 [10974] LOG:  listening on IPv4 address "127.0.0.1", port 5432
2017-11-05 15:56:27.208 +03 [10974] LOG:  listening on Unix socket "/tmp/.s.PGSQL.5432"
2017-11-05 15:56:27.228 +03 [10975] LOG:  database system was shut down at 2017-11-05 15:54:28 +03
2017-11-05 15:56:27.231 +03 [10974] LOG:  database system is ready to accept connections
2017-11-05 15:56:54.091 +03 [10991] LOG:  [Failed Password] Username: root Password: dert
2017-11-05 15:56:54.091 +03 [10991] FATAL:  password authentication failed for user "root"
2017-11-05 15:56:54.091 +03 [10991] DETAIL:  Role "root" does not exist.
    Connection matched pg_hba.conf line 86: "host    all            all            0.0.0.0/0          password"
2017-11-05 15:56:54.094 +03 [10990] LOG:  incomplete startup packet
2017-11-05 15:56:54.095 +03 [10992] LOG:  [Failed Password] Username: root Password: mert
2017-11-05 15:56:54.095 +03 [10992] FATAL:  password authentication failed for user "root"
2017-11-05 15:56:54.095 +03 [10992] DETAIL:  Role "root" does not exist.
    Connection matched pg_hba.conf line 86: "host    all            all            0.0.0.0/0          password"
2017-11-05 15:56:54.096 +03 [10989] LOG:  incomplete startup packet
2017-11-05 15:56:54.099 +03 [10994] LOG:  [Failed Password] Username: root Password: test
2017-11-05 15:56:54.099 +03 [10994] FATAL:  password authentication failed for user "root"
2017-11-05 15:56:54.099 +03 [10994] DETAIL:  Role "root" does not exist.
    Connection matched pg_hba.conf line 86: "host    all            all            0.0.0.0/0          password"
2017-11-05 15:56:54.103 +03 [10993] LOG:  incomplete startup packet
root@ubuntu:~/honeypot#
```

```

GNU nano 2.5.3  File: /root/honeypot/vsftpd-3.0.3/prelogin.c
if (tunable_userlist_enable)
{
    int located = str_contains_line(&p_sess->userlist_str, &p_sess->user_str);
    if ((located && tunable_userlist_deny) || !located && !tunable_userlist_deny))
    {
        check_login_delay();
        vsf_cstdio_write(p_sess, FTP_LOGINERR, "Permission denied.");
        check_login_fails(p_sess);
        str_empty(&p_sess->user_str);
        return;
    }
}
if (is_anon && tunable_no_anon_password)
{
    /* Fake a password */
    str_alloc_text(&p_sess->ftp_arg_str, "<no password>");
    handle_pass_command(p_sess);
}
else
{
    vsf_cstdio_write(p_sess, FTP_GIVEPASSWORD, "Please specify the password.");
}
}

static void
handle_pass_command(struct vsf_session* p_sess)
{
if (str_isempty(&p_sess->user_str))
{
    vsf_cstdio_write(p_sess, FTP_NEEDUSER, "Login with USER first.");
    return;
}

    /* Mert SARICA */
    struct mystr str_log_line = INIT_MYSTR;
    str_append_text(&str_log_line, "[Failed Password] client: ");
    str_append_str(&str_log_line, &p_sess->remote_ip_str);
    str_append_text(&str_log_line, " user: ");
    str_append_str(&str_log_line, &p_sess->user_str);
    str_append_text(&str_log_line, " Password: ");
    str_append_str(&str_log_line, &p_sess->ftp_arg_str);
    vsf_log_line(p_sess, kVSFLogEntryConnection, &str_log_line);

    /* These login calls never return if successful */
    if (tunable_one_process_mode)
    {
        vsf_one_process_login(p_sess, &p_sess->ftp_arg_str);
    }
    else
    {
        vsf_two_process_login(p_sess, &p_sess->ftp_arg_str);
        vsf_cstdio_write(p_sess, FTP_LOGINERR, "Login incorrect.");
        check_login_fails(p_sess);
        str_empty(&p_sess->user_str);
        /* FALLTHRU if login fails */
    }
}

static void check_login_delay()
{
if (tunable_delay_failed_login)
{
    vsf_sysutil_sleep((double) tunable_delay_failed_login);
}
}

static void check_login_fails(struct vsf_session* p_sess)
{
    str_append_text(p_str, "o ");

    /* Access mode: anonymous/real user, and identity */
    if (p_sess->is_anonymous && !p_sess->is_guest)
    {
        str_append_text(p_str, "a");
        str_append_str(p_str, &p_sess->anon_pass_str);
    }
    else
    {
        if (p_sess->is_guest)
        {
            str_append_text(p_str, "g ");
        }
        else
        {
            str_append_text(p_str, "r ");
        }
        str_append_str(p_str, &p_sess->user_str);
    }
    str_append_char(p_str, ',');
    /* Service name, authentication method, authentication user id */
    str_append_text(p_str, "ftp 0 ");
    /* completion status */
    if (succeeded)
    {
        str_append_char(p_str, 'c');
    }
    else
    {
        str_append_char(p_str, 'i');
    }
}

    /* Mert SARICA */
static void
vsf_log_do_log_vsftpd_format(struct vsf_session* p_sess, struct mystr* p_str,
                             int succeeded, enum EVSFLogEntryType what,
                             const struct mystr* p_log_str)
{
    str_empty(p_str);
    if (!tunable_syslog_enable)
    {
        /* Date - vsf_sysutil_get_current_date updates cached time */
        str_append_text(p_str, vsf_sysutil_get_current_date());
    }
    if (!str_isempty(p_log_str))
    {
        str_append_text(p_str, ",");
        str_append_str(p_str, p_log_str);
        str_append_char(p_str, ',');
    }
}

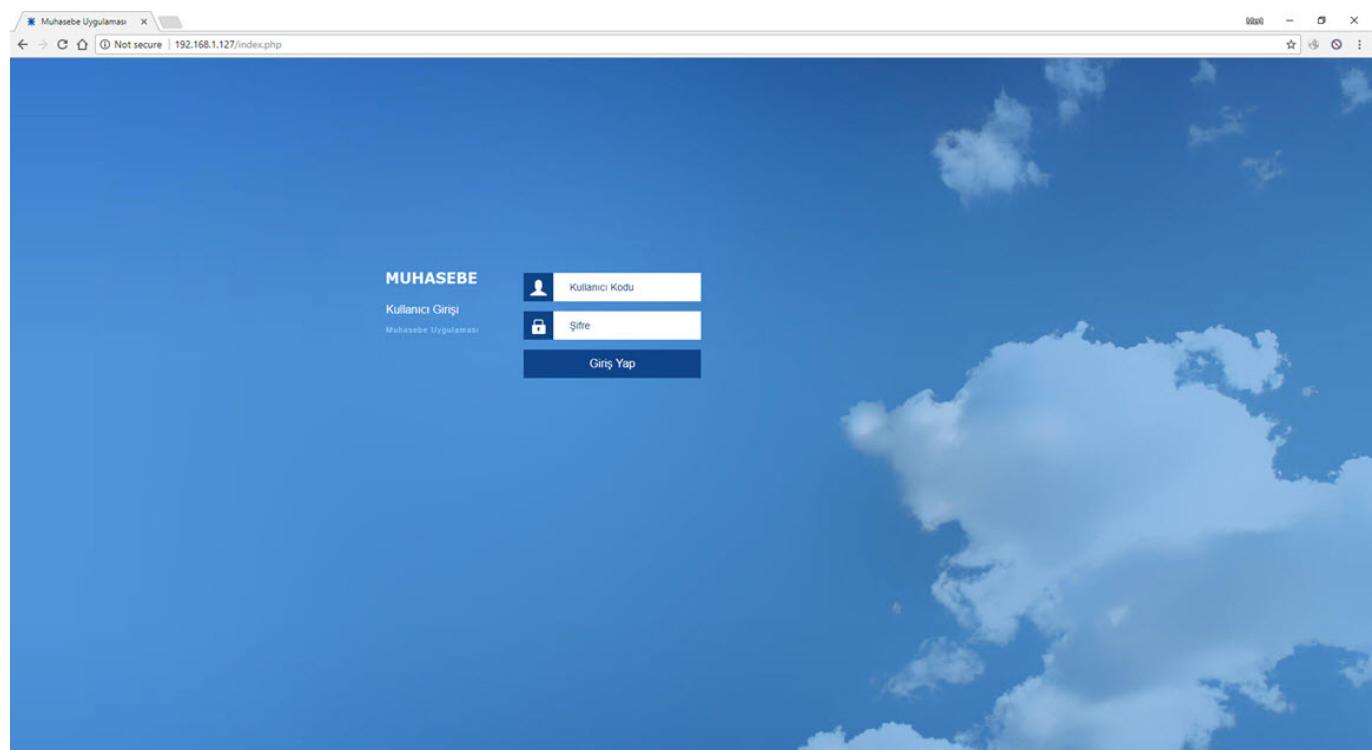
```

```
✓ Batcave | ✓ Batcave (1) | ✓ Batcave (3) | ✓ Batcave (2) x | ✓ Batcave (4)
root@ubuntu:/usr/local/pgsql/data# ftp 127.0.0.1
Connected to 127.0.0.1.
220 Muhasebe sunucusu.
Name (127.0.0.1:root): mert
331 Please specify the password.
Password:
530 Login incorrect.
Login failed.
ftp>
```



```
✓ Batcave | ✓ Batcave (1) x | ✓ Batcave (3) | ✓ Batcave (2) | ✓ Batcave (4)
root@ubuntu:/var/log# cat vsftpd.log
Sun Nov  5 17:12:15 2017: client "127.0.0.1"
Sun Nov  5 17:12:18 2017: client "127.0.0.1", "[Failed Password] Username: mert Password: dert"
Sun Nov  5 17:12:18 2017: client "127.0.0.1"
root@ubuntu:/var/log#
```

Son olarak ise web sunucusunun ana klasöründe oluşturduğum index.php dosyasının, yapılan hatalı giriş denemelerini kayıt altına almasını sağladım.



GNU nano 2.5.3 File: index.php

```

</script>
return true;
}

</body>
</html>

```

<?php

```

if($_REQUEST) {
    // echo $_POST['username'];
    // echo $_POST['password'];

    if(isset($_POST['username']) && isset($_POST['password'])) {
        ob_start();
        $txt = print_r($_POST, true);
        // echo $txt;

        $re1='^(([^\\r\\n]*)*)'; # Non-greedy match on filler
        $re2='(([^\\r\\n]*)*)'; # Square Braces 1
        $re3='(([^\\r\\n]*)*)'; # Any Single Character 1
        $re4='(([^\\r\\n]*)*)'; # Square Braces 2
        $re5='(([^\\r\\n]*)*)'; # Any Single Character 3

        if (preg_match_all ("/.",$re1,$re2,$re3,$re4,"is", $txt, $matches)) {
            error_log(print_r(date('d-m-Y H:i:s', $_SERVER['REQUEST_TIME']), true), "[Failed Password] Client: ".print_r($_SERVER['REMOTE_ADDR'], true)."\n".print_r($_POST['username'], true)." Password: ".print_r($_POST['password'], true)."\n", 3, "/var/log/apache_form.log");
        }
        ob_end_flush();
    } else {
        html();
    }
} else {
    html();
}

```

Get Help Write Out Where Is Cut Text Justify Cur Pos Prev Page First Line Where Is Next Mark Text Indent Text Undo Exit Read File Replace Uncut Text To Spell Go To Line Next Page Last Line To Bracket Copy Text Unindent Text Redo

2041

Batcave X | Batcave (1) | Batcave (3) | Batcave (2) | Batcave (4) | Kali (VM)

```

root@ubuntu:/var/www/html# cat /var/log/apache_form.log
07-11-2017 20:40:48 [Failed Password] Client: 192.168.1.144 Username: mert Password: dert
root@ubuntu:/var/www/html#

```

Python ile geliştirdiğim loginmon isimli aracı da saat başı /var/log klasörü altında yer alan ve açık halde saklanan parolaları, hatalı giriş denemesi yapan kullanıcı adları, ip adresleri ve tarihler ile alıp, Splunk'a gönderecek şekilde hazırladım.

```

GNU nano 2.5.3
File: /etc/cron.hourly/loginmon

#!/usr/bin/python
# Failed Login Attempt Monitor v1.0
# Author: Mert SARICA
# E-mail: mert [ . ] sarica [ @ ] gmail [ . ] com
# URL: https://www.mertsarica.com

import time
import os
import datetime
import hashlib
import sys
import smtplib
from email.header import Header
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from email.mime.base import MIMEBase
from email import encoders

debug = 0

# Log Files
logfile = "/var/log/passmon.txt"
ftppdlog = "/var/log/vftpd.log"
authlog = "/var/log/auth.log"
pglog = "/var/log/pgsql.log"
forilog = "/var/log/httpd_error.log"
forilog = "/var/log/apache_form.log"

# E-mail Parameters
sender = "mertsarica@gmail.com"
recipient = "mert.sarica@gmail.com"
server = "127.0.0.1"
port = 25

def send_data():
    try:
        FILE = open(logfile,"rb")
    except Exception,e:
        if debug:
            print "[*] send_data error: " + str(e)
        return
    subject = "Failed Login Attempts"
    msg = MIMEMultipart()
    msg.set_charset("iso-8859-9")
    msg["Subject"] = str(Header(subject, "iso-8859-9"))
    msg["To"] = recipient
    part = MIMEBase("application", "octet-stream")
    part.set_payload(FILE.read())
    encoders.encode_base64(part)
    part.add_header('Content-Disposition', 'attachment; filename="passmon.txt"')
    # Attach parts into message container.
    # According to RFC 2046, the last part of a multipart message, in this case
    # the HTML message, is best and preferred.
    msg.attach(part)
    try:
        session = smtplib.SMTP(server, port)
        session.ehlo()
        session.sendmail(sender, recipient, msg.as_string())
    except Exception,e:
        if debug:
            print "[*] send_email error: " + str(e)

GNU nano 2.5.3
File: /etc/cron.hourly/loginmon

rg = re.compile(re1=re2+re3+re4+re5+re6+re7+re8+re9+re10+re11+re12,re.IGNORECASE|re.MULTILINE)
for m in rg.finditer(txt):
    if m:
        date1=m.group(1)
        mydate = datetime.datetime.strptime(date1, '%d-%m-%Y')
        time1=m.group(2)
        slos=m.group(3)
        var1=m.group(4)
        ipaddress1=m.group(5)
        word1=m.group(6)
        var2=m.group(7)
        word2=m.group(8)
        var3=m.group(9)
        line = "date:" + mydate.strftime('%d %b') + " Time:" + time1 + " Service:WEBFORM IP:" + ipaddress1 + " Username:" + var2 + " Password:" + var3
        print "[*] " + line
        log(line)

def banner():
    os.system("clear")
    print "="*70
    print "Failed Login Attempt Monitor - https://www.mertsarica.com"
    print "="*70

def main():
    while 1==1:
        try:
            print "[*] analyzing logs..."
            parse_authlog()
            parse_ftppdlog()
            parse_pglog()
            parse_httalog()
            parse_formlog()

        except KeyboardInterrupt:
            banner()
            print "[*] Bye..."
            sys.exit(1)
        except Exception,e:
            print "[*] Error: " + str(e)
            sys.exit(1)

if __name__ == '__main__':
    banner()
    main()

=====
[!] Batcave (1) | Batcave (3) | Batcave (2) | Batcave (4) | Kali (VM)
=====
Failed Login Attempt Monitor - https://www.mertsarica.com
=====
[*] Analyzing Logs...
[+] Date:7 Nov Time:20:03:07 Service:SSH IP:192.168.1.144 Username:root Password:dert
[+] Date:7 Nov Time:20:03:07 Service:SSH IP:192.168.1.144 Username:root Password:mert
[+] Date:7 Nov Time:20:03:07 Service:SSH IP:192.168.1.144 Username:root Password:test
[+] Date:5 Nov Time:17:53:23 Service:FTP IP:192.168.1.144 Username:mert Password:dert
[+] Date:9 Nov Time:22:19:35 Service:PGSQL IP:192.168.1.144 Username:root Password:mert
[+] Date:9 Nov Time:22:19:35 Service:PGSQL IP:192.168.1.144 Username:root Password:test
[+] Date:9 Nov Time:22:19:35 Service:PGSQL IP:192.168.1.144 Username:root Password:dert
[+] Date:8 Nov Time:20:16:10 Service:HTACCESS IP:192.168.1.144 Username:mert Password:dert URL:/admin/
[+] Date:8 Nov Time:20:16:12 Service:HTACCESS IP:192.168.1.144 Username:mert Password:dert URL:/admin/
[+] Date:7 Nov Time:20:40:48 Service:WEBFORM IP:192.168.1.144 Username:mert Password:dert

```

```

Batcave X
root@ubuntu:/var/log# head -n 15 passmon.txt
Date:4-12-17 Time:17:18:10 Service:SSH IP:59.63.166.83 Username:root Password:1.0
Date:4-12-17 Time:16:35:44 Service:SSH IP:59.63.166.83 Username:root Password:root!@#
Date:5-12-17 Time:19:16:21 Service:SSH IP:58.242.83.25 Username:root Password:jumpstart
Date:4-12-17 Time:16:35:48 Service:SSH IP:59.63.166.83 Username:root Password:root!@#
Date:4-12-17 Time:16:35:49 Service:SSH IP:59.63.166.83 Username:root Password:pumpkin
Date:5-12-17 Time:19:16:22 Service:SSH IP:58.242.83.25 Username:root Password:jun123
Date:6-12-17 Time:09:23:45 Service:SSH IP:42.7.26.16 Username:root Password:qweasdqwe
Date:4-12-17 Time:16:35:50 Service:SSH IP:59.63.166.83 Username:root Password:planet1
Date:6-12-17 Time:09:23:46 Service:SSH IP:42.7.26.16 Username:root Password:qweasdzx!@#
Date:5-12-17 Time:19:16:23 Service:SSH IP:58.242.83.25 Username:root Password:jussi
Date:4-12-17 Time:16:35:50 Service:SSH IP:59.63.166.83 Username:root Password:powerpc
Date:6-12-17 Time:09:23:47 Service:SSH IP:42.7.26.16 Username:root Password:qazwsxedc!@#123
Date:5-12-17 Time:19:16:24 Service:SSH IP:58.242.83.25 Username:root Password:justdoit
Date:6-12-17 Time:09:23:48 Service:SSH IP:42.7.26.16 Username:root Password:!#@qwe!@#
Date:5-12-17 Time:19:16:25 Service:SSH IP:58.242.83.25 Username:root Password:kaiser
root@ubuntu:/var/log#

```

```

root@ubuntu:~# /opt/splunkforwarder/bin/splunk add monitor /var/log/passmon.txt -sourcetype Loginmon -index main
Added monitor of '/var/log/passmon.txt'.
root@ubuntu:~#

```

Yaklaşık 1 ay sonunda Splunk üzerinde biriken ~500.000 kayıt üzerinde gerçekleştirmiş olduğum analizde, sistemime gerçekleştirilen deneme yanlış ve sözlük saldırısının en çok Çin'den gerçekleştirildiğini, en fazla saldırısı alan servisin SSHD olduğunu, kullanıcı adı olarak en çok root, parola için ise 1234 ile deneme yapıldığını öğrenmiş oldum.

| List | | Format | 20 Per Page | |
|---|--|--------|-------------|-------|
| <input type="checkbox"/> Hide Fields | <input checked="" type="checkbox"/> All Fields | I | Time | Event |
| > 12/9/17 Date:9-12-17 Time:15:32:31 Service:FTP IP:195.158.26.114 Username:www-data Password:test | | | | |
| > 12/8/17 Date:8-12-17 Time:11:22:37 Service:FTP IP:117.199.85.149 Username:admin Password:backward | | | | |
| client_country | | | | |
| 72 Values, 78.18% of events | | | | |
| Reports | | | | |
| Top values Top values by time Rare values | | | | |
| Events with this field | | | | |
| Top 10 Values Count % | | | | |
| China 378,886 95.429% | | | | |
| Portugal 6,098 1.536% | | | | |
| United States 2,799 0.709% | | | | |
| Ukraine 2,194 0.552% | | | | |
| France 1,835 0.462% | | | | |
| Singapore 780 0.196% | | | | |
| Italy 511 0.129% | | | | |
| Asia/Pacific Region 508 0.128% | | | | |
| India 490 0.123% | | | | |
| Korea, Republic of 321 0.081% | | | | |
| > 12/8/17 Date:8-12-17 Time:06:32:35 Service:SSH IP:123.183.209.139 Username:root Password:iloveme | | | | |
| 6.32:35:00 AM date = 8/12/17 host = ubuntu ip = 123.183.209.139 password2 = iloveme service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:32:35 username = root | | | | |
| > 12/8/17 Date:8-12-17 Time:06:32:04 Service:SSH IP:123.183.209.139 Username:root Password:123qwe!@# | | | | |
| 6.32:04:00 AM date = 8/12/17 host = ubuntu ip = 123.183.209.139 password2 = 123qwe!@# service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:32:04 username = root | | | | |
| > 12/8/17 Date:8-12-17 Time:06:32:03 Service:SSH IP:123.183.209.139 Username:root Password:pld | | | | |
| 6.32:03:00 AM date = 8/12/17 host = ubuntu ip = 123.183.209.139 password2 = pld service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:32:03 username = root | | | | |
| > 12/8/17 Date:8-12-17 Time:06:32:03 Service:SSH IP:123.183.209.139 Username:root Password:antinea | | | | |
| 6.32:03:00 AM date = 8/12/17 host = ubuntu ip = 123.183.209.139 password2 = antinea service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:32:03 username = root | | | | |
| > 12/8/17 Date:8-12-17 Time:06:31:22 Service:SSH IP:123.183.209.139 Username:root Password:nelson | | | | |
| 6.31:22:00 AM date = 8/12/17 host = ubuntu ip = 123.183.209.139 password2 = nelson service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:31:22 username = root | | | | |
| > 12/8/17 Date:8-12-17 Time:06:31:22 Service:SSH IP:123.183.209.139 Username:root Password:banban | | | | |
| 6.31:22:00 AM date = 8/12/17 host = ubuntu ip = 123.183.209.139 password2 = banban service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:31:22 username = root | | | | |
| > 12/8/17 Date:8-12-17 Time:06:31:21 Service:SSH IP:123.183.209.139 Username:root Password:sesson | | | | |
| 6.31:21:00 AM date = 8/12/17 host = ubuntu ip = 123.183.209.139 password2 = session service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:31:21 username = root | | | | |

| Selected Fields | | Time | | | Event | |
|-------------------------------|---|------------|--------------------|-------------|--|--|
| Selected Fields | <input checked="" type="checkbox"/> Hide Fields <input type="checkbox"/> All Fields | | | | | |
| <code>service</code> | | | | | | |
| 3 Values, 100% of events | | Selected | Yes | No | ne:www-data Password:test
service = FTP source = /var/log/passmon.txt sourcetype = LogInmon time = 15:32:31 username = www-data | |
| Reports | | Top values | Top values by time | Rare values | me:admin Password:backnrd
ame:admin Password:admin
e:admin Password:password
me:admin Password:default
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 07:06:47 username = admin | |
| Events with this field | | | | | admin Password:12345
ice = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:48:30 username = admin | |
| Values | Count | % | | | Date: 8-12-17 Time: 06:48:30 Service:SSH IP:121.14.7.244 Username:admin Password:password
Date: 8-12-17 Time: 06:48:29 Service:SSH IP:121.14.7.244 Username:admin Password:7uYKk0admin
Date: 8-12-17 Time: 06:48:29 Service:SSH IP:121.14.7.244 Username:admin Password:default
Date: 8-12-17 Time: 06:48:29 Service:SSH IP:121.14.7.244 Username:admin Password:root
Date: 8-12-17 Time: 06:48:29 Service:SSH IP:121.14.7.244 Username:root Password:12345
Date: 8-12-17 Time: 06:48:29 Service:SSH IP:121.14.7.244 Username:root Password:kokos
Date: 8-12-17 Time: 06:32:36 Service:SSH IP:123.183.209.199 password2=kokos
Date: 8-12-17 Time: 06:32:36 Service:SSH IP:123.183.209.199 password2=juna1234
Date: 8-12-17 Time: 06:32:36 Service:SSH IP:123.183.209.199 password2=luna1234
Date: 8-12-17 Time: 06:32:36 Service:SSH IP:123.183.209.199 password2=123qwe123
Date: 8-12-17 Time: 06:32:36 Service:SSH IP:123.183.209.199 password2=pwd1
Date: 8-12-17 Time: 06:32:36 Service:SSH IP:123.183.209.199 password2=session
Date: 8-12-17 Time: 06:32:36 Service:SSH IP:123.183.209.199 password2=session | |
| Interesting Fields | | | | | date=_day
date=_month
date=_second
date=_year
date=_hour
date=_minute
date=_month_2
date=_second_50
date=_year_1
date=_zone_1
index_1
inexact_1
punct_100+
sparkline_server_1
timendpos_2
timestamppos_1 | |
| 45 more fields | | | | | | |
| + Extract New Fields | | | | | | |
| | | | | | | |
| Selected Fields | | Time | | | Event | |
| Selected Fields | <input checked="" type="checkbox"/> Hide Fields <input type="checkbox"/> All Fields | | | | | |
| <code>username</code> | | | | | ne:www-data Password:test
service = FTP source = /var/log/passmon.txt sourcetype = LogInmon time = 15:32:31 username = www-data | |
| >100 Values, 99.99% of events | | Selected | Yes | No | me:admin Password:backnrd
ame:admin Password:admin
e:admin Password:password
me:admin Password:default
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 07:06:47 username = admin | |
| Reports | | Top values | Top values by time | Rare values | ice = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:48:30 username = admin | |
| Events with this field | | | | | admin Password:12345
ice = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:48:29 username = admin
admin Password:password
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 07:06:43 username = admin
me:admin Password:root
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 07:06:39 username = admin
admin Password:root
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:48:29 username = admin
admin Password:kokos
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:32:36 username = root
admin Password:juna1234
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:32:36 username = root
admin Password:luna1234
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:32:36 username = root
admin Password=123qwe123
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:32:04 username = root
admin Password:session
service = SSH source = /var/log/passmon.txt sourcetype = LogInmon time = 06:31:21 username = root | |
| Top 10 Values | Count | % | | | Date: 8-12-17 Time: 06:31:22 Service:SSH IP:123.183.209.199 Username:root Password:nelson
Date: 8-12-17 Time: 06:31:22 Service:SSH IP:123.183.209.199 password2=nelson
Date: 8-12-17 Time: 06:31:22 Service:SSH IP:123.183.209.199 Username:root Password:banban
Date: 8-12-17 Time: 06:31:21 Service:SSH IP:123.183.209.199 Username:root Password:session
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=pi
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=root
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=antinea
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=user
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=test
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=guest
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=centos
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=support
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=ubuntu
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=pi
Date: 8-12-17 Time: 06:32:03 Service:SSH IP:123.183.209.199 password2=administrator | |
| Interesting Fields | | | | | date=_day
date=_month
date=_second
date=_year
date=_hour
date=_minute
date=_month_2
date=_second_50
date=_year_1
date=_zone_1
index_1
inexact_1
punct_100+
sparkline_server_1
timendpos_2
timestamppos_1 | |
| 45 more fields | | | | | | |
| + Extract New Fields | | | | | | |
| | | | | | | |

| password2 | | | | | |
|---|-------|--------|------------|--|---|
| < Hide Fields | | | All Fields | | |
| >100 Values, 100% of events | | | | Selected | Yes <input checked="" type="checkbox"/> No <input type="checkbox"/> |
| Reports | | | | | |
| Top values Top values by time Rare values | | | | Events with this field | |
| Top 10 Values | | | | | |
| | Count | % | | | |
| 1234 | 2,136 | 0.421% | | | |
| admin | 1,925 | 0.379% | | | |
| root | 1,453 | 0.286% | | | |
| 123456 | 1,107 | 0.218% | | | |
| password | 987 | 0.194% | | | |
| support | 942 | 0.185% | | | |
| ubnt | 929 | 0.183% | | | |
| 12345 | 899 | 0.177% | | | |
| raspberry | 714 | 0.141% | | | |
| default | 709 | 0.14% | | | |
| More Top Values Load All Reset | | | | | |
| More Reports More Fields Extract New Fields | | | | | |
| <i>me:mm-data Password: test
ice = FTP source = /var/log/passmon.txt sourcetype = Loginmon time = 15:32:31 username = www-data
me:admin Password: backward
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 11:22:37 username = admin
ame:admin Password:admin
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 07:06:47 username = admin
e:admin Password:password
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 07:06:43 username = admin
me:admin Password:default
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 07:06:39 username = admin
admin Password:12345
ice = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:30 username = admin
admin Password:password
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:30 username = admin
admin Password:7u7kcoadmin
n : service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:29 username = admin
admin Password:Juna1234
vice = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:29 username = admin
admin Password:default</i> | | | | | |
| Show all X | | | | | |

| ip | | | | | |
|---|---------|---------|------------|--|---|
| < Hide Fields | | | All Fields | | |
| >100 Values, 100% of events | | | | Selected | Yes <input checked="" type="checkbox"/> No <input type="checkbox"/> |
| Reports | | | | | |
| Top values Top values by time Rare values | | | | Events with this field | |
| Top 10 Values | | | | | |
| | Count | % | | | |
| 59.63.166.63 | 129,438 | 25.49% | | | |
| 42.7.26.15 | 78,242 | 15.408% | | | |
| 59.242.83.26 | 39,284 | 7.736% | | | |
| 61.177.172.66 | 39,201 | 7.732% | | | |
| 61.177.172.60 | 29,975 | 5.903% | | | |
| 218.87.109.154 | 19,342 | 3.809% | | | |
| 42.7.26.16 | 18,373 | 3.618% | | | |
| 61.177.172.10 | 14,859 | 2.926% | | | |
| 59.63.186.36 | 14,374 | 2.831% | | | |
| 59.242.83.33 | 13,912 | 2.74% | | | |
| More Top Values Load All Reset | | | | | |
| More Reports More Fields Extract New Fields | | | | | |
| <i>me:mm-data Password: test
ice = FTP source = /var/log/passmon.txt sourcetype = Loginmon time = 15:32:31 username = www-data
me:admin Password: backward
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 11:22:37 username = admin
ame:admin Password:admin
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 07:06:47 username = admin
e:admin Password:password
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 07:06:43 username = admin
me:admin Password:default
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 07:06:39 username = admin
admin Password:12345
ice = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:30 username = admin
admin Password:password
service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:30 username = admin
admin Password:7u7kcoadmin
n : service = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:29 username = admin
admin Password:Juna1234
vice = SSH source = /var/log/passmon.txt sourcetype = Loginmon time = 06:48:29 username = admin
admin Password:default</i> | | | | | |
| Show all X | | | | | |

Bir sonraki yazıda görüşmek dileğiyle herkese güvenli günler dilerim.