

HACKING LABORATORY INFORMATION MEDICAL SYSTEMS FOR FUN AND PROFIT

ORIGINAL PROBLEM

Too many COVID test results

- Lab was multiple months behind on results
- Humans could not enter them fast enough
- Lab was trying to automate the process

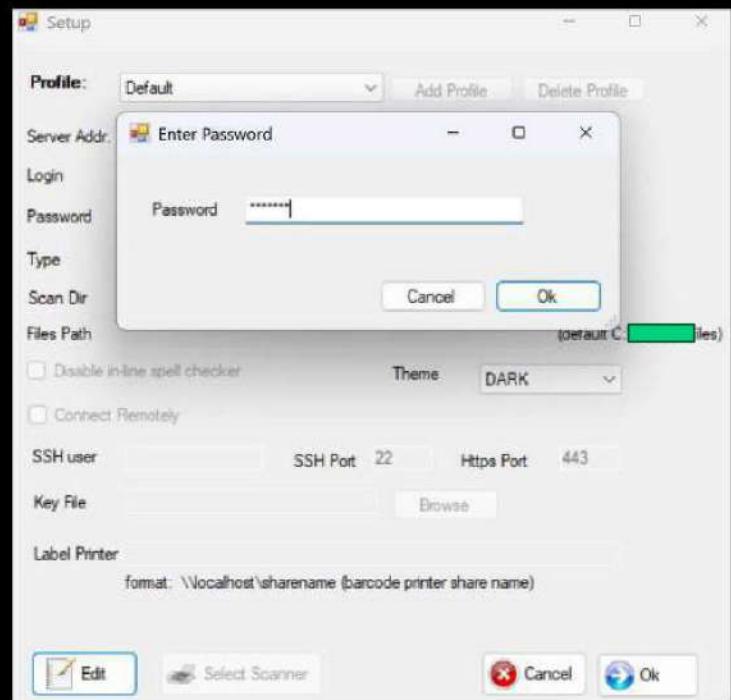
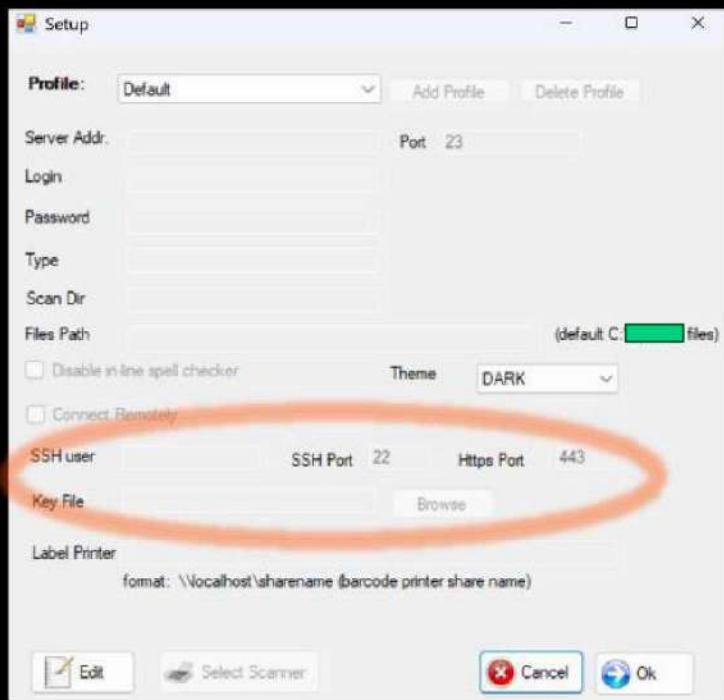
Initial request

- Setup a farm of computers running windows
- Install Microsoft Power Automate
- Catch up results so humans can take over

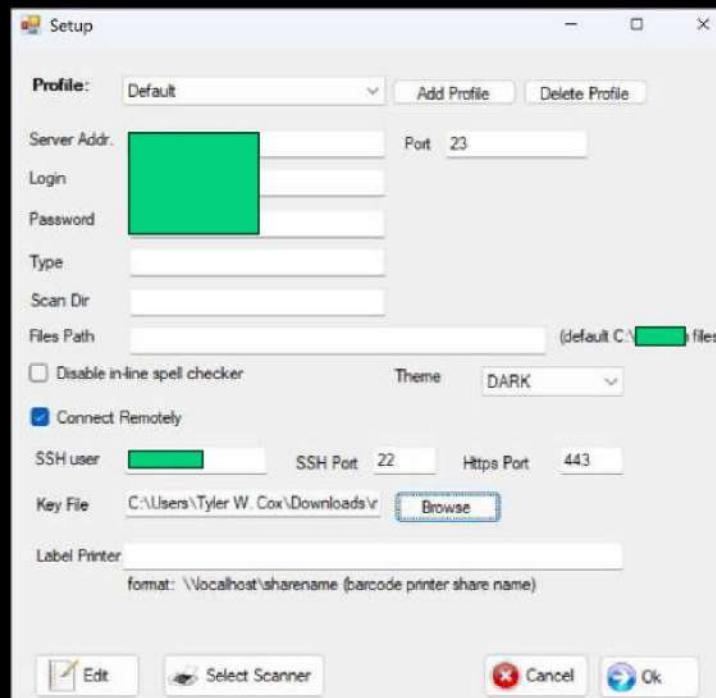
SETUP INSTRUCTIONS

- Download [REDACTED] client at [https://\[REDACTED\]/%20setup.3.0.7.exe](https://[REDACTED]/%20setup.3.0.7.exe)
- Install Labgen client
 - Since this is not a Windows program, you may be prompted to bypass "safety" messages to install.
- Run [REDACTED]
- Go to Setup
- Click Edit
- Enter PW: [REDACTED]
- Enter Server Address [REDACTED]
- Enter Port 23
- Enter the T-number for Login (see below)
- Enter monitor^ for Password
- Click Connect Remotely
- Enter SSH User = [REDACTED]
- SSH Port = 22
- Browse to extracted file and select for key file
- Click OK
- Restart [REDACTED]
- Login with username, hit F2 and create password
- Login with Username and New Password
- Call if you need a hand

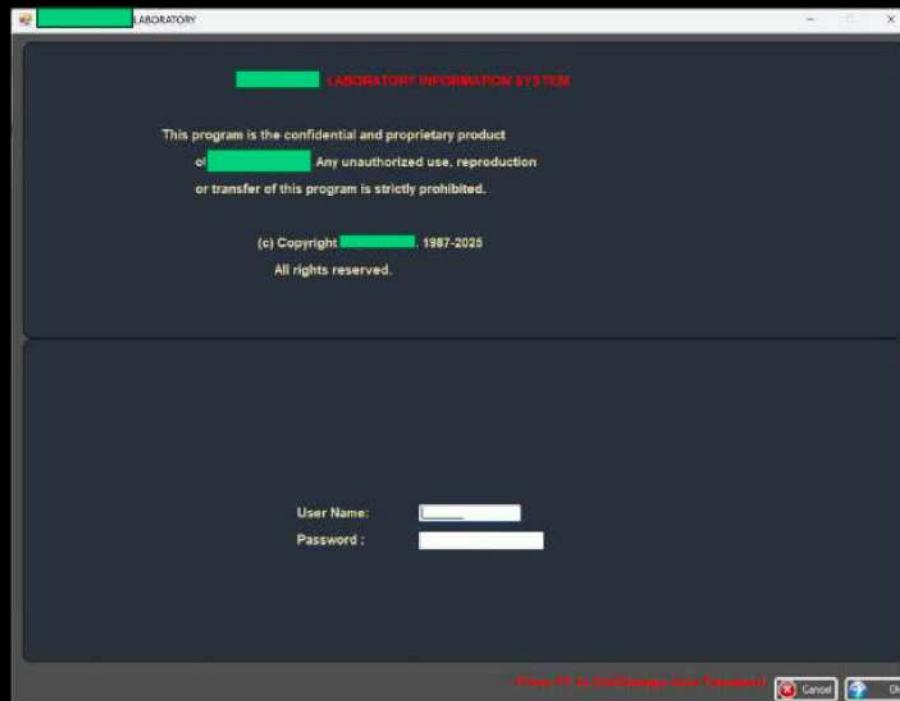
STATIC PASSWORD BAKED IN



WELL KNOWN PORTS AND KEY FILE



LOOKS A LOT LIKE A DUMB TERMINAL



NEW PROBLEM

- Software has built in rate limiting “to prevent human error”
- Power Automate Script needed about 6 minutes per record
- Math says a PC can’t do more than 250 results in a day
- Software only supports 50 logins
- Max throughput is capped at an optimistic 12,500 results daily
- New tests entering the pipeline at 20,000 a day

NO ACCESS

```
C:\Users\tcox>ssh [REDACTED].net
ssh: connect to host [REDACTED].net port 22: Connection timed out
```

I SPY WITH MY LITTLE MITM

Unlock complete network capture on Windows and power up server-side inspection with Reverse Proxy - all in Fiddler Everywhere now! Give them a try!

Telerik Fiddler Classic

OVERVIEW FIDDLER TOOLS DOCS & SUPPORT TRY FOR FREE

Fiddler Classic

A Web Debugging Proxy Tool for Windows

A Windows-only tool that logs HTTP(s) network traffic.

Try For Free* Compare With Fiddler Everywhere

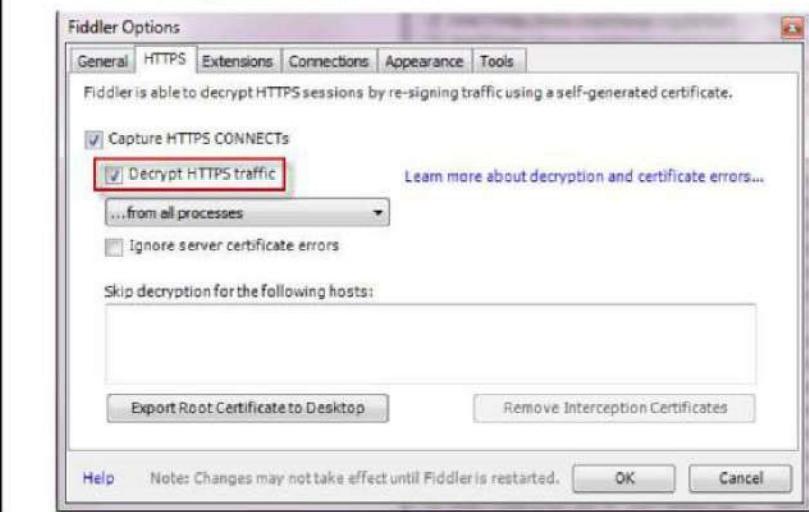
*Please be aware that Fiddler Classic is not in active development and offers no commitments for releases, patches or tech support. By using this product, you assume all associated risks. We recommend upgrading to [Fiddler Everywhere](#).



ENABLE HTTPS DECRYPTION

Enable HTTPS traffic decryption

1. Click Tools > Options > HTTPS.
2. Click the Decrypt HTTPS Traffic box.

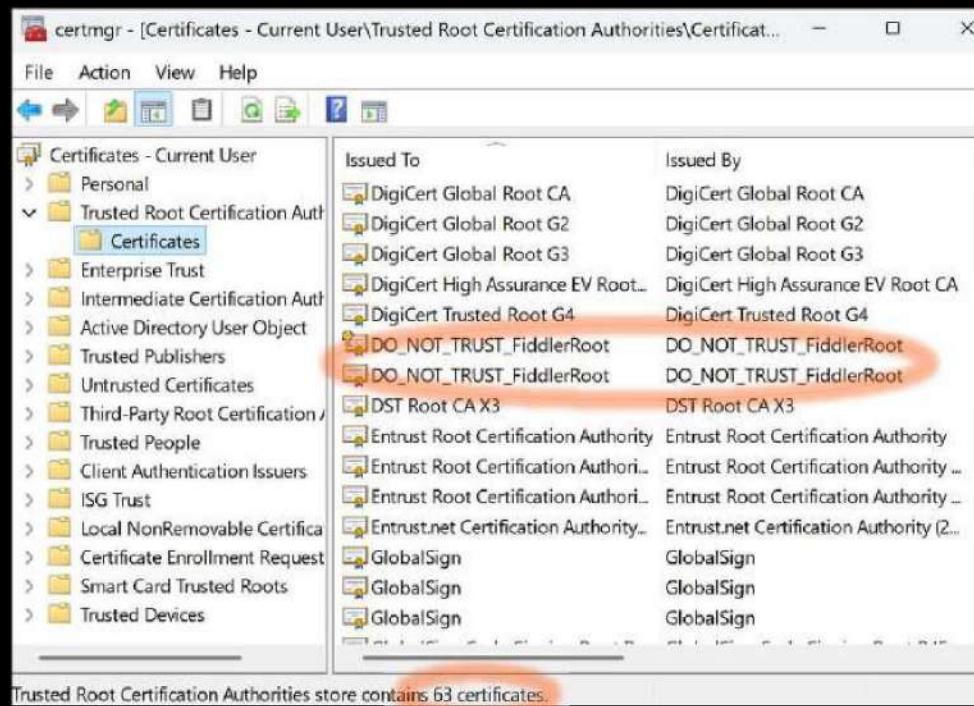


CAN'T SAY WE WEREN'T WARNED

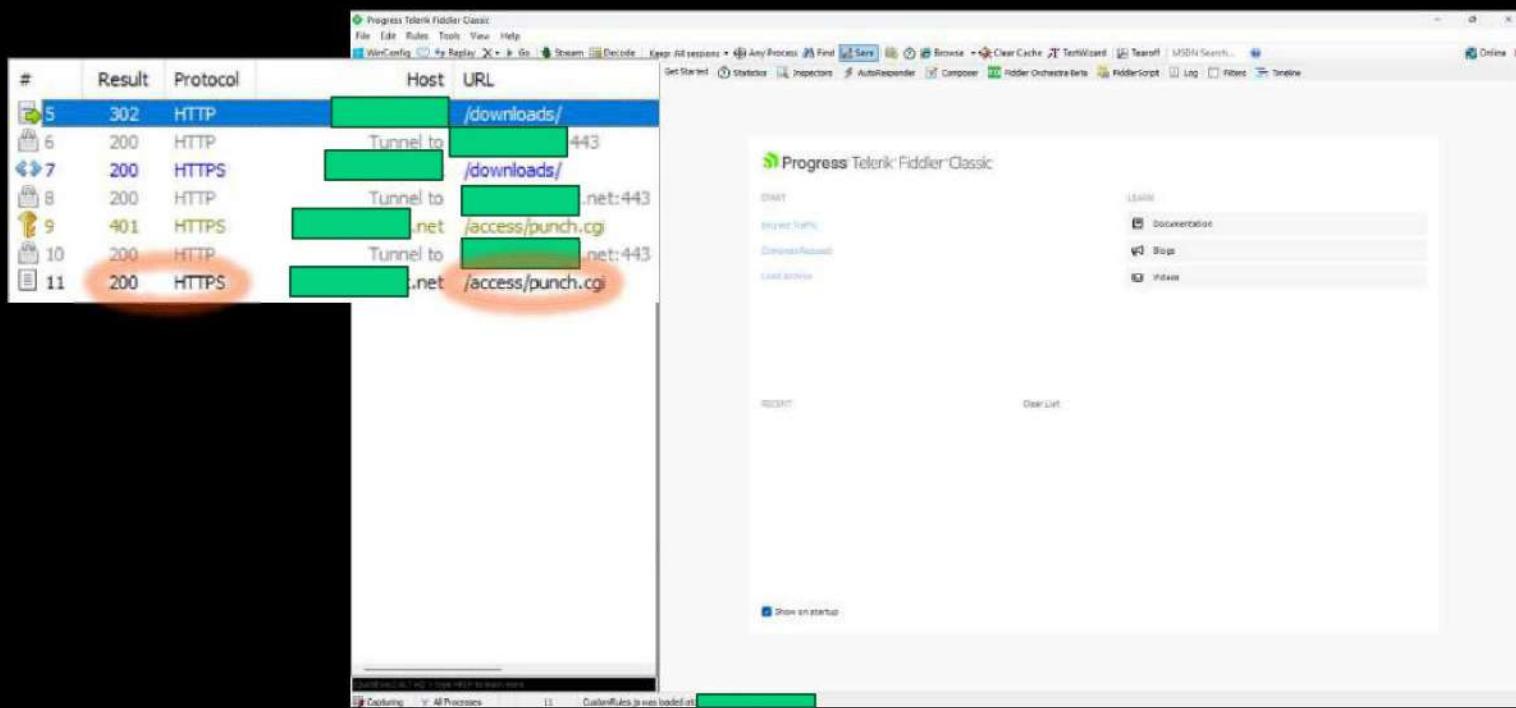
The image is a collage of four screenshots from Fiddler Classic, showing various stages of certificate configuration and errors.

- Top Left:** A warning message titled "SCARY TEXT AHEAD: Read Carefully!" stating: "To intercept HTTPS traffic, Fiddler Classic generates a unique root certificate." It includes a note: "You may configure Windows to trust this root certificate to suppress security warnings. This is generally safe." Below it is a button: "Trust the Fiddler Classic Root certificate? Yes No".
- Top Right:** A "Security Warning" dialog box. It says: "You are about to install a certificate from a certification authority (CA) claiming to represent: DO_NOT_TRUST_FiddlerRoot". It continues: "Windows cannot validate that the certificate is actually from 'DO_NOT_TRUST_FiddlerRoot'. You should confirm its origin by contacting 'DO_NOT_TRUST_FiddlerRoot'. The following number will assist you in this process: Thumbprint (sha1): 37FE6B5D 5E690E22 2DF2E45B 83AD6CD1 77121EDB". It includes a "Warning:" section: "If you install this root certificate, Windows will automatically trust any certificate issued by this CA. Installing a certificate with an unconfirmed thumbprint is a security risk. If you click X acknowledge this risk." At the bottom are "Yes" and "No" buttons.
- Bottom Left:** A "TrustCert Success" dialog box with the message: "Added Fiddler Classic's root certificate to the Machine Root List." It has an "OK" button.
- Bottom Right:** A "Certificate Error" dialog box. It shows a warning icon and the text: "Session #6: The server [REDACTED] presented a certificate that did not validate, because it was issued to a different host." It displays certificate details: SAN *.lab *.lab.com *.lab.s.net, SUBJECT: CN = [REDACTED]. It includes an "Ignore errors (unsafe) and proceed?" button with "Yes" and "No" options.

WINDOWS IS NOW VERY TRUSTING



LOOK AT THAT TRAFFIC



ANOTHER STATIC PASSWORD

Get Started Statistics Inspectors AutoResponder Composer Fiddler Orchestra Beta FiddlerScript Log Filters Timeline

Headers TextView SyntaxView WebForms HexView Auth Cookies Raw JSON XML

Request Headers

GET /access/punch.cgi HTTP/1.1

Security

Authorization: Basic Z3Vlc3Q6YmExbmEybmEz

Transport

Connection: Close

Host: [REDACTED].net

Transformer Headers TextView SyntaxView ImageView

No Proxy-Authenticate Header is present.

No WWW-Authenticate Header is present.

Get Started Statistics Inspectors AutoResponder Composer Fiddler Orchestra Beta FiddlerScript Log Filters Timeline

Headers TextView SyntaxView WebForms HexView Auth Cookies Raw JSON XML

No Proxy-Authorization Header is present.

Authorization Header is present: Basic Z3Vlc3Q6YmExbmEybmEz

Decoded Username:Password= guest:baina2na3

Transformer Headers TextView SyntaxView ImageView HexView WebView Auth Caching Cookies Raw JSON XML

No Proxy-Authenticate Header is present.

No WWW-Authenticate Header is present.

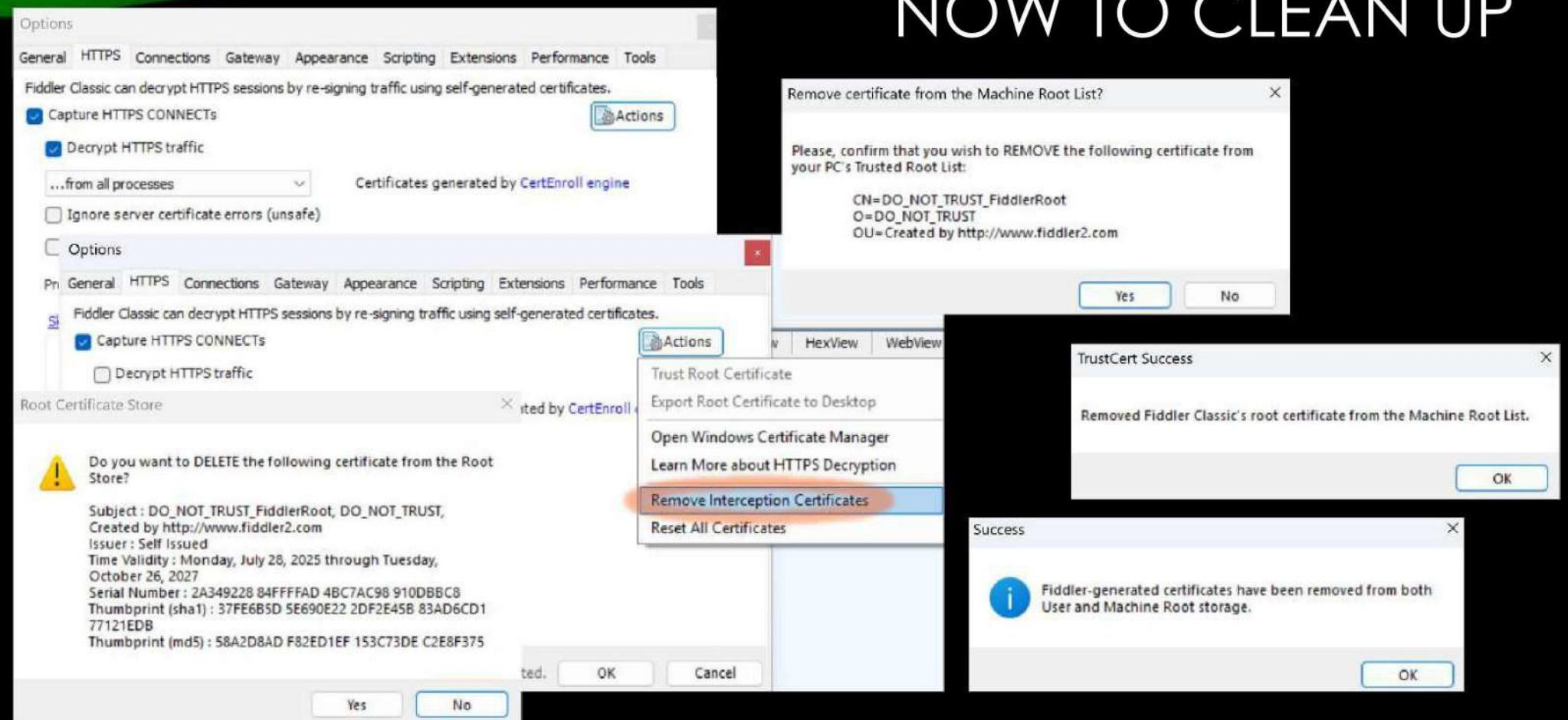
[REDACTED]

[REDACTED]

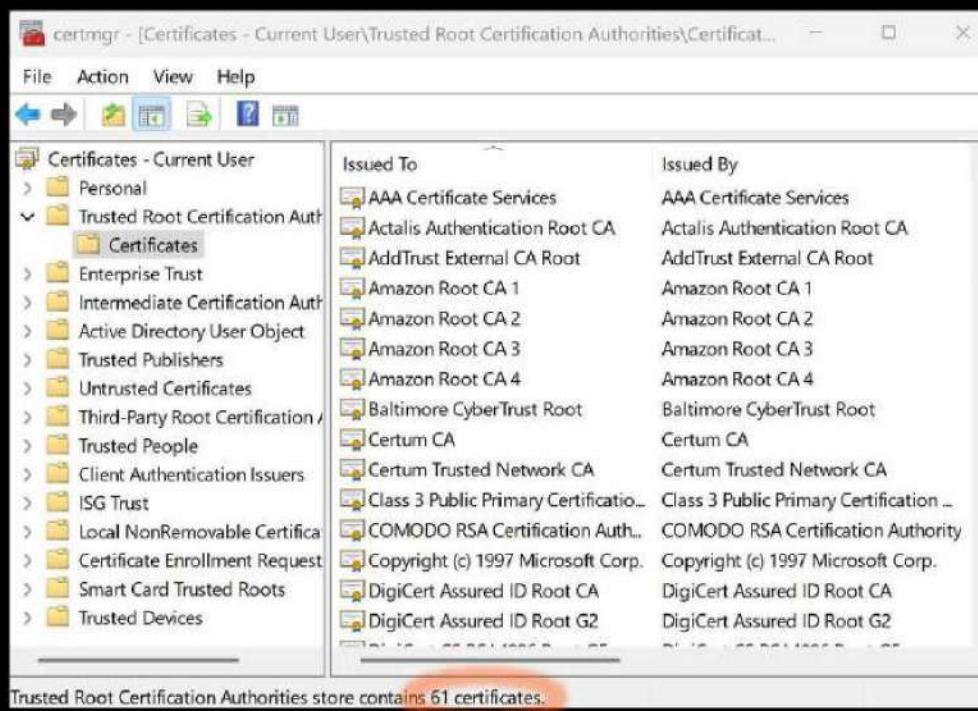
MAY I PLEASE HAVE SOME SSH?

```
C:\Users\Tyler W. Cox>ssh [REDACTED].net
The authenticity of host '[REDACTED].net ([REDACTED])' can't be established.
ED25519 key fingerprint is SHA256:[REDACTED].
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? |
```

NOW TO CLEAN UP



VERIFY THEY'RE GONE



CAN WE PUNCH A HOLE FROM THE CLI?

```
administrator@mrqlinuxauto:~$ curl -s -k -i --raw "https://[REDACTED].net/access/punch.cgi" -H "Authorization: Basic Z3Vlc3Q6YmExbmEybmEz" -H "Host: [REDACTED].net" -H "Connection: Close"
HTTP/1.1 200 OK
Date: [REDACTED]
Server: Apache
Strict-Transport-Security: max-age=63072000; includeSubDomains
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
Connection: close
Transfer-Encoding: chunked
Content-Type: text/plain; charset=UTF-8

5a
Access through the firewall has been temporarily provided for your IP Address: [REDACTED]

35
Your access will expire at [REDACTED]

0
```

LET'S ENCAPSULATE THAT

```
knock.sh
#!/bin/bash
#does a https knock to open port 22 on [REDACTED]
>this touches a file called lastknock to track and only knock once an hour
|
#if the file isn't there create it with a date in the past
if [ ! -f ~/lastknock ]
then
  » touch -t 202002020202.02 ~/lastknock
fi
|
#check the age and then either knock and update or exit
if [ $(date +%s)-$(stat --c "%Y" ~/lastknock) ] -lt 3600 ]
then
  » echo knocked less than an hour ago
else
  » echo knocking again
  » curl -s -k -i --raw "https://[REDACTED].net/access/punch.cgi" --H "Authorization: Basic Z3Vlc3Q6YmExbmEybmEz" --H "Host: [REDACTED].net" --H "Connection: Close" | grep expire
  » touch ~/lastknock
fi
```

HOW ABOUT REAL SSH

```
administrator@mrqlinuxauto:~$ ssh [REDACTED]@[REDACTED].net -i ~/ss  
h/id_[REDACTED]_rsa  
Last login: [REDACTED] from [REDACTED]
```

```
[REDACTED].net - CentOS Linux release 7.9.2009 (Core)  
website: [REDACTED].net  
local-dns: [REDACTED].net  
local-ip: 172.24.0.100 - 10.123.102.88  
ovpn-dns: [REDACTED].me  
Public ssh: 22
```

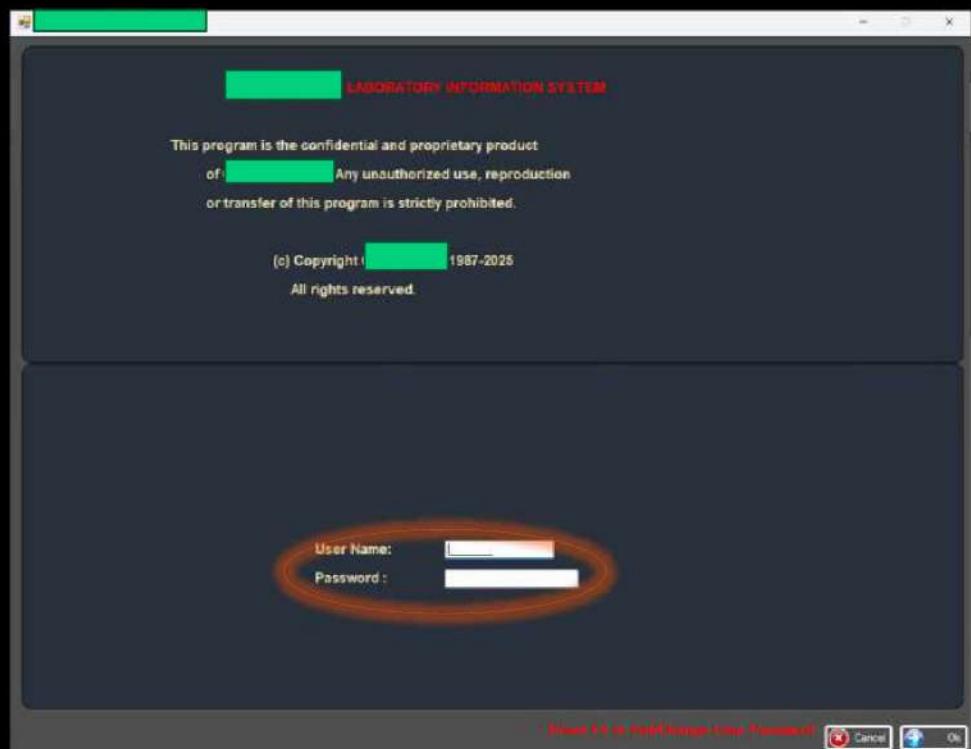
Licenses: 155 concurrent
Last st-number issued is: st-155

```
[REDACTED]@[REDACTED] ~]$
```

THAT WASN'T THE TERMINAL WE EXPECTED TO BE IN...THINK OLDER

```
administrator@mrqlinuxauto:~$ ssh [REDACTED]@[REDACTED].net -i ~/s:  
h/id_[REDACTED]_rsa  
Last login: [REDACTED] from [REDACTED]  
  
[REDACTED].net - CentOS Linux release 7.9.2009 (Core)  
website: [REDACTED].net  
local-dns: [REDACTED].net  
local-ip: 172.24.0.100 - 10.123.102.88  
ovpn-dns: [REDACTED].me  
Public ssh: 22  
  
Licenses: 155 concurrent  
Last st-number issued is: st-155  
  
[REDACTED] ~]$ telnet localhost  
Trying 127.0.0.1...  
Connected to localhost.  
Escape character is '^]'.  
  
CentOS Linux release 7.9.2009 (Core)  
Linux [REDACTED].net 3.10.0-1160.105.1.el7.x86_64 #1 SMP Thu Dec  
7 15:39:45 UTC 2023 x86_64 x86_64 x86_64 GNU/Linux  
16 CPU(s) detected (via ACPI) at Speed: 2095 MHz with Cache: 22528  
KB  
31987312 kB of system memory visible to OS.  
eth0: 172.24.0.100/24  
172.24.2.100/24 - 00:15:5d:00:04:05  
  
[REDACTED] login:
```

THAT LOOKS FAMILIAR



WHAT'S REALLY HAPPENING IN THE TUNNEL IS VT100

```

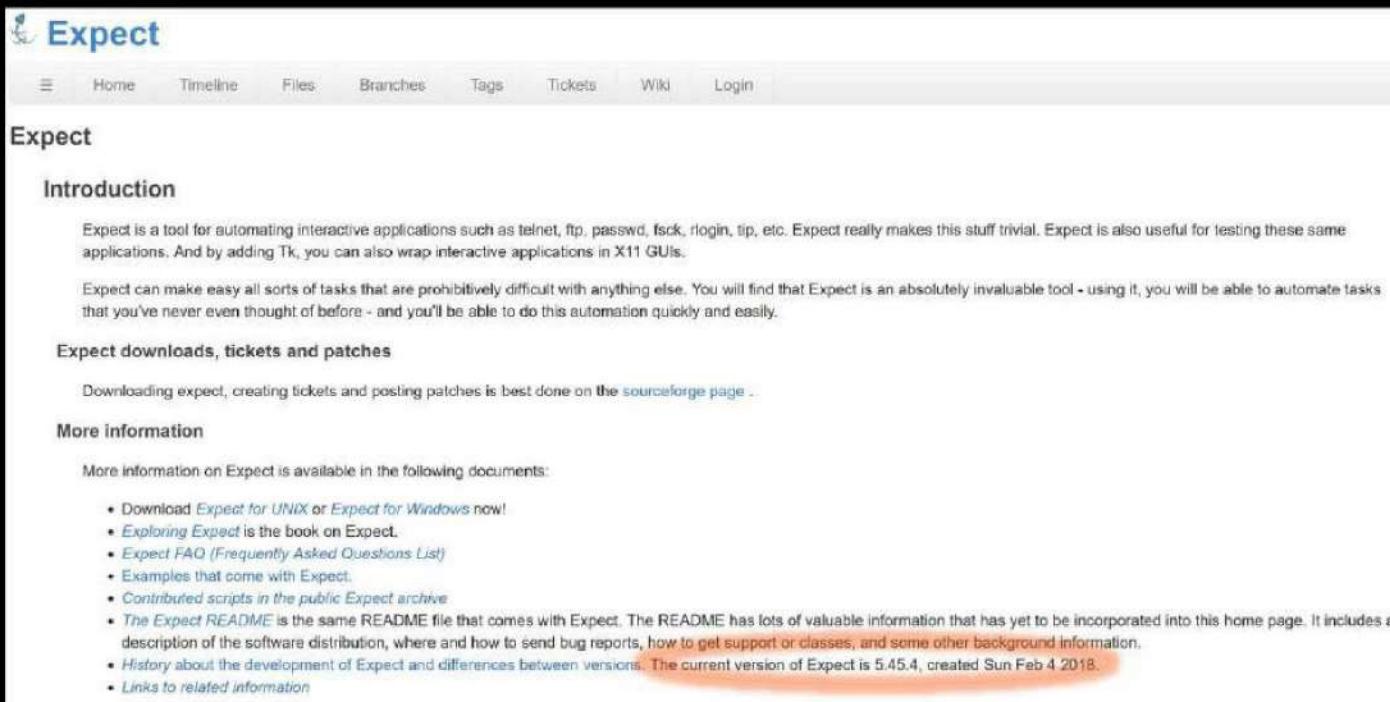
[[1;104H][1;45H](0|m[41][?7H][H][2]
[[2;79H]
[[3;79H]
[[4;79H]
[[5;79H]
[[6;79H]
[[7;79H]
[[8;79H]
[[9;79H]
[[10;79H]
[[11;79H]
[[12;79H]
[[13;79H]
[[14;79H]
[[15;79H]
[[16;79H]
[[17;79H]
[[18;79H]
[[19;79H]
[[20;79H]
[[21;79H]
[[22;79H]
[[23;79H]
[[24;79H][H-----[[11;80H
[[24d]-----[[11;80H
-- Any unauthorized use, reproduction or transfer of this program is strictly prohibited. Copyright [REDACTED] 1987-2023[9;20HAll
proprietary product[[5;16Hof [REDACTED] rights reserved.[12;22H(B)4) address P2 to Set/Change User Password[[6;18H(c) Copyright [REDACTED] 1987-2023[9;20HAll
[[13;25HUser Name: [[18;45H][29;25HPassword: [[18;47H][18;57H
***[[45;20H]10491
[[11;20H][1040H][1145H][0|16|4]127H[[14;22H
[[11;6;7m LABORATORY SYSTEM copyright 1987
[[12;76H]
[[13;30H(B)[n [3;76H(0|0;7m
[[4;24H(WARN MSG) [4;76H(0|0;7m.
User
  a) Worksheets[[8;55H][8;24H
  b) Enter Results[[6;55H][6;24H
  v) Verify/Approve Results[[7;55H][10;24H
  c) Call panic result[[11;55H][11;24H
  p) Patient Report[[12;55H][12;24H
  t) Patient Receipt[[13;55H][13;24H
  A) Patients Received[[14;55H][14;24H
  f) Patient File Maintenance[[15;55H][15;24H
  S) System Dictionary[[16;55H][16;24H
  U) Utilities & Options[[17;55H][17;24H
  H) Interface menu[[18;55H][18;24H
  L) Print Barcode Label[[19;55H][19;24H
  X) Check Remote Send Status [[20;24H
  Q) Quit from system[[21;55H][21;24H[[24;37H#PS+Log-out[[7;25H((0|0;2m) Requisition Entry
[[0|m[7;24H][45;10H]10491

```

(S)(n[7;24H R) Requisition Entry[[0|7;55H][7;24H ((0|0;2m) Requisition Entry

setwin DECSTBM	Set top and bottom lines of a window ^[[cv>;cv>	
cursorup(n) CUU	Move cursor up n lines ^[[cn>A	
cursordn(n) CUD	Move cursor down n lines ^[[cn>B	
cursorrt(n) CUF	Move cursor right n lines ^[[cn>C	
cursorlf(n) CUB	Move cursor left n lines ^[[cn>D	
cursorhome	Move cursor to upper left corner ^[[;H	
cursorhere	Move cursor to upper left corner ^[[;H	
cursorpos(v,h) CUP	Move cursor to screen location v,h ^[[cv>;ch>H	
hvhome	Move cursor to upper left corner ^[[f	
hvhere	Move cursor to upper left corner ^[[f	
hvpos(v,h) CUP	Move cursor to screen location v,h ^[[cv>;ch>f	
index IND	Move/scroll window up one line ^[[D	
revindex RI	Move/scroll window down one line ^[[M	
nextline NEL	Move to next line ^[[E	
savecursor DECSC	Save cursor position and attributes ^[[7	
restorecursor DECSC	Restore cursor position and attributes ^[[8	
tabset HTS	Set a tab at the current column ^[[H	
tabclr TBC	Clear a tab at the current column ^[[g	
tabclr TBC	Clear a tab at the current column ^[[0g	
tabclrall TBC	Clear all tabs ^[[3g	
dhtop DECDHL	Double-height letters, top half ^[[#3	
dhtbot DECDHL	Double-height letters, bottom half ^[[#4	
swsh DECSWL	Single width, single height letters ^[[#5	
dwsh DECDWL	Double width, single height letters ^[[#6	
clearol EL0	Clear line from cursor right ^[[K	
clearol EL0	Clear line from cursor right ^[[ek	
clearbol EL1	Clear line from cursor left ^[[1K	
clearline EL2	Clear entire line ^[[2K	
cleareos EDO	Clear screen from cursor down ^[[J	
cleareos EDO	Clear screen from cursor down ^[[0J	
clearbos ED1	Clear screen from cursor up ^[[1J	
clearscreen ED2	Clear entire screen ^[[2J	
devstat DSR	Device status report ^[[5n	
termok DSR	Response: terminal is OK ^[[0n	
termnok DSR	Response: terminal is not OK ^[[3n	
getcursor DSR	Get cursor position ^[[6n	
cursorpos CPR	Response: cursor is at v,h ^[[cv>;ch>R	

OLD PROBLEMS REQUIRE OLD SOLUTIONS



The screenshot shows the homepage of the Expect project on SourceForge. The page has a header with a logo, navigation links for Home, Timeline, Files, Branches, Tags, Tickets, Wiki, and Login. The main content area is titled "Expect" and contains sections for "Introduction", "Expect downloads, tickets and patches", and "More information". The "Introduction" section describes Expect as a tool for automating interactive applications and mentions Tk integration. The "Expect downloads, tickets and patches" section links to SourceForge. The "More information" section lists various resources like the book, FAQ, examples, and README file.

Expect

Home Timeline Files Branches Tags Tickets Wiki Login

Expect

Introduction

Expect is a tool for automating interactive applications such as telnet, ftp, passwd, fsck, login, tip, etc. Expect really makes this stuff trivial. Expect is also useful for testing these same applications. And by adding Tk, you can also wrap interactive applications in X11 GUIs.

Expect can make easy all sorts of tasks that are prohibitively difficult with anything else. You will find that Expect is an absolutely invaluable tool - using it, you will be able to automate tasks that you've never even thought of before - and you'll be able to do this automation quickly and easily.

Expect downloads, tickets and patches

Downloading expect, creating tickets and posting patches is best done on the sourceforge page ..

More information

More information on Expect is available in the following documents:

- Download *Expect for UNIX* or *Expect for Windows* now!
- Exploring Expect* is the book on Expect.
- Expect FAQ (Frequently Asked Questions List)*
- Examples that come with Expect.
- Contributed scripts in the public Expect archive*
- The Expect README* is the same README file that comes with Expect. The README has lots of valuable information that has yet to be incorporated into this home page. It includes a description of the software distribution, where and how to send bug reports, how to get support or classes, and some other background information.
- History about the development of Expect and differences between versions*. The current version of Expect is 5.45.4, created Sun Feb 4 2018.
- Links to related information*

BUILT ON OLDER SOLUTIONS

Expect is an extension to the [Tcl](#) scripting language written by Don Libes.^[3] The program automates interactions with programs that expose a [text terminal](#) interface. Expect, originally written in 1990 for the [Unix](#) platform, has since become available for [Microsoft Windows](#) and other systems.

Basics [edit]

Expect is used to automate control of interactive [applications](#) such as [Telnet](#), [FTP](#), [passwd](#), [fsck](#), [rlogin](#), [tip](#), [SSH](#), and others.^[4] Expect uses [pseudo terminals](#) (Unix) or emulates a console (Windows), starts the target program, and then communicates with it, just as a human would, via the terminal or console interface.^[5] [Tk](#), another Tcl extension, can be used to provide a [GUI](#).^[6]

Expect

Original author(s)	Don Libes
Developer(s)	Nils Carlson
Stable release	5.45.4 ^[1] / 4 February 2018; 7 years ago
Written in	C
Operating system	POSIX, Windows
License	Public domain ^[2]
Website	core.tcl-lang.org/expect/ ^[3]

BUILT FOR REALLY OLD PROBLEMS

Tcl (pronounced "tickle" or "TCL";^[8] originally **Tool Command Language**) is a [high-level](#), general-purpose, interpreted, dynamic programming language. It was designed with the goal of being very simple but powerful.^[9] Tcl casts everything into the mold of a [command](#), even programming constructs like variable assignment and procedure definition.^[10] Tcl supports multiple [programming paradigms](#), including [object-oriented](#), [imperative](#), [functional](#), and [procedural](#) styles.

It is commonly used embedded into [C](#) applications,^[11] for rapid prototyping, scripted applications, GUIs, and testing.^[12] Tcl interpreters are available for many [operating systems](#), allowing Tcl code to run on a wide variety of systems. Because Tcl is a very compact language, it is used on [embedded systems](#) platforms, both in its full form and in several other small-footprint versions.^[13]

The popular combination of Tcl with the Tk extension is referred to as **Tcl/Tk** (pronounced "tickle teak"^[citation needed] or "tickle TK") and enables building a [graphical user interface](#) (GUI) natively in Tcl. Tcl/Tk is included in the standard [Python](#) installation in the form of [Tkinter](#).

History [edit]

Tcl	
 <i>Tcl</i> ⁹	Multi-paradigm: event-driven, functional, imperative, object-oriented
Designed by	John Ousterhout
Developer	Tcl Core Team ^[1]
First appeared	1988; 37 years ago
Stable release	9.0.2 ^[2] / 2 July 2025; 29 days ago
Typing discipline	Dynamic typing, everything is a string
Implementation language	C, Tcl
License	BSD-style ^[3]
Filename extensions	.tcl, .tbc ^[4]
Website	www.tcl-lang.org

WHEN ALL YOU HAVE IS A HAMMER

checkout.php X

```
D: > tcx > Desktop > stage > [REDACTED] > www > tyler > checkout.php
1 <?php require_once $_SERVER['DOCUMENT_ROOT'] . "/Connections/SQL.php"; ?>
2 <?php
3 echo sqlsrv_array($SQL, "
4 set nocount on
5 declare @out as datetime
6 set @out=getdate();
7
8 declare @order as varchar(max)
9
10 select top 1
11     @order=tyler_accession
12 from
13     [REDACTED] dbo.tyler
14 where
15     tyler_checkout is null
16 order by
17     newid()
18
19 update
20     marquisvm.dbo.tyler
21 set
22     tyler_checkout=@out
23 where
24     tyler_accession=@order
25 set nocount off
26 select @order as ordernum
27 ")[0]["ordernum"];
28 ?>
```

mrqlinuxauto.marquisvm - dbo.tyler -> X SQL

Column Name	Data Type	Allow Nulls
tyler_accession	varchar(10)	<input type="checkbox"/>
tyler_checkout	datetime	<input checked="" type="checkbox"/>
tyler_checkin	datetime	<input checked="" type="checkbox"/>

checkin.php X

```
D: > tcx > Desktop > stage > [REDACTED] > www > tyler > checkin.php
1 <?php require_once $_SERVER['DOCUMENT_ROOT'] . "/Connections/SQL.php"; ?>
2 <?php
3 echo sqlsrv_array($SQL, "
4 set nocount on
5
6 declare @order as varchar(max)
set @order=?
7
8 update
9     [REDACTED] dbo.tyler
10 set
11     tyler_checkin=getdate()
12 where
13     tyler_accession=@order
14 set nocount off
15 select @order as ordernum
16 ",array(
17     param("ordernum")
18 )
19 )
20 )[0]["ordernum"];
21 ?>
```

LET'S BUILD A BIG LOOP

```
tyler.tcl
#!/usr/bin/expect3
#exp_internal=1
#Number of orders to run
set force_conservative 0 ;# set to 1 to force conservative mode even if
# > > script wasn't run conservatively originally
if {!$force_conservative} {
    set send_slow {1.1}
    proc send {ignore arg} {
        sleep 1
        exp_send -s $arg
    }
}
set maxLoop 100
set timeout 10
spawn ssh [REDACTED] net-i-/ssh/id_[REDACTED].rsa
expect '*[REDACTED]*~(.send "telnet localhost\r")'
expect '*[REDACTED]* login:*(*)'
send "root\r"
expect "*Password*:*(send "[REDACTED]\r")"
expect "User-Name":()
expect "Password":*(send "[REDACTED]\r")"
expect "Log-out":*(send "rm")
for {set z 0} {$z<$maxLoop} {incr z} {
    set clientID [exec wget -qO- http://localhost/tyler/checkout.php] | tee -a log.txt
    if {$clientID == ""} {
        puts "\x1b\[H\x1b\[2JNo More Records"
        exit 1
    }
    set blank [exec echo "\r" >> log.txt]
    set output [exec echo $clientID > lastfile]
    send_user $clientID
    send_record $clientID
    puts "\x1b\[H\x1b\[2J$clientID"
    set finishedClientID [exec wget -qO- http://localhost/tyler/checkin.php?ordernum=$clientID]
}
```

WITH SPAGHETTI INSIDE

```

proc send_record {::cLID::} {
    set timeout 3
    expect "Accession-Number" {
        > send "$cLID\t"
    } timeout
    after 500
    send "\033^E"
    after 500
    expect "Accession-Number" {
        > send "$cLID\t"
    } timeout
    > send user:"!!!!NOACCESSION-$cLID!!!!"
    > exit 1
    }
}

expect {
    "PCRM" {
        > expect {
            > > -re {^\x1b[4;62H(.*)}\n
            > > after 500
            > > send "\033^E"
            > > }
            > default {
                > > send "\033\\[B\\033\\[BNegative"
                > > after 350
                > > send "\033\\[18~"
                > > after 350
                > > send "\033^E"
            }
        }
    }
}

"LG03" {
    > expect {
        > > -re {^\x1b[4;62H(.*)}\n
        > > after 500
        > > send "\033^E"
        > > }
        > default {
            > > send "\033\\[B\\033\\[BNegative"
            > > after 350
            > > send "\033\\[18~"
            > > after 350
            > > send "\033^E"
        }
    }
}

default {
    > > after 500
    > > send "\033^E"
}
}

```

PLEASE SIR COULD I HAVE SOME THREADS?

```
clear¶
seq·400·|·xargs·-P·10·-n·1·~/accession.tcl¶
seq·500·|·xargs·-P·10·-n·1·~/fullaccession.tcl¶
seq·700·|·xargs·-P·10·-n·1·~/negative.tcl¶
seq·30·|·xargs·-P·4·-n·1·~/positive.tcl¶
```



PROFIT???