Fine penetration tests for fine websites

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TunnelBear Security Assessment Summary 07.2017

Cure53, Dr.-Ing. Mario Heiderich & Team

Appendix

Bug ID	Component	Description
Medium Severity (14)		
TB-04-011	API	Attacker could abuse friend email invite system by including a domain name in their invite message that would be automatically turned into a link by some clients.
TB-03-028	Android/Mac	Android and Mac clients based some protections on hard-coded secrets. These secrets could be extractable from the official release binaries. In addition to this, an insecure algorithm was used to protect passwords at rest on Android. Finally, less than ideal cryptographic algorithms were chosen to protect data at rest.
TB-03-025	Android/iOS/ Mac/Win	Client apps used HTTP links, prone to MitM attacks, to certain external resources, such as help articles.

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TB-03-024	Android/iOS/ Mac/Win	TunnelBear clients failed to fully leverage certificate pinning to protect TLS communications. In the case of Android and Mac OS, this affected only requests outside of the TunnelBear API, such as AWS requests that download configuration information. In the case of the iOS and Windows clients, however, there was no pinning whatsoever. This could allow malicious adversaries with a certificate trusted by the OS store (most governments, some companies) to intercept and modify network communications.
TB-03-027	Android	Android app failed to mitigate TapJacking attacks. This could allow malicious apps without any special permissions to render an overlay, launch the TunnelBear app in the background, and attempt to fool a user into performing actions on the TunnelBear app.
TB-03-023	Android	It was found that the Android app failed to leverage the available operating system protections to avoid data leakage through screenshots. A malicious app with screen recording or root permissions could steal data through screenshots.

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TB-03-006	Misc	RADIUS authentication server running on the test-server validated all authentications through a GET function. All GET requests sent by this method could end up in web server log files where they are stored in plain-text. If an attacker were able to compromise TunnelBear's API servers and read the web server logs, s/he would gain usernames and passwords of accounts that were authenticated using RADIUS.
TB-03-014	Server Hardening	Most Linux default installations have several security options disabled due to requiring individual work or possibly affecting the system's usability for the majority of users. There are several configuration options available that are proven to significantly raise the security-bar for a Linux server.
TB-03-013	Server Hardening	TunnelBear VPN servers were found to be running *vanilla* Linux kernels, which are historically known to be easy to exploit as soon as a vulnerability is found. It was recommended to consider either switching to another distribution that supports hardened kernels, or, alternatively, to compile a *vanilla* kernel with the *Grsecurity* patch-set in a custom manner.

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TB-03-017	Server-Hardening	The audit identified several configuration files with weak permission settings. In the event a VPN server were compromised, they would give an attacker new information about the underlying system and potentially disclose further attack vectors.
TB-03-018	Server-Hardening	One of the users in *sudoers* file was found to be able to escalate to root without providing a password. Even though this is a restricted user, if it was compromised, the attacker would not have the need to exploit further security holes to escalate privileges.
TB-03-003	VPN	It was found that the test server is running an outdated version of OpenVPN and OpenSSL. It was also found that the configuration files fail to enforce the use of TLS version 1.2. There are multiple downgrade attacks that can be used when running TLS version 1.0. If they were to be successfully exploited, the encryption and integrity of the VPN tunnel could be severely undermined.
TB-03-034	iOS/Mac/ Extension	It was found that iOS and browser extension clients initiate VPN connections based on the reply of a DNS query. A malicious attacker with the ability to spoof DNS requests could leverage this weakness to prevent legitimate TunnelBear users from connecting to the VPN

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TB-03-022	iOS	It was found that the iOS application partially failed to take advantage of the native iOS file system protections and did not fully protect some of its data-files at rest.
Low Severity		
TB-03-001	Server Hardening	Some files on were found to have unmapped user and group attached; which could server as a source of pivot if system were to be compromised. *dbus-daemon* was running under the *messagebus* user with *setuid-bit* set. This would allow the attacker to gain root if *messagebus* user was compromised.
TB-03-002	Server Hardening	There were adjustments proposed to /etc/sysctl.conf file for some additional network hardening.
TB-03-008	Web	It was found that the TunnelBear website (www.tunnelbear.com) used session cookies scoped to the parent domain (tunnelbear.com). An attacker with the ability to modify network communications could set up a phishing attack targeting a TunnelBear user.
TB-03-012	Web	One of the API endpoints supported redirects via an URL parameter. No restrictions were in place, which made it vulnerable to open redirects.

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TB-03-016	Backend	It was found that OpenVPN authentication script was mis-configured in such a way that it would save a temporary file with credentials in the current working directory instead of temporary directory at an in-memory location.
TB-03-020	Web Backend	The library used to parse diagnostic reports submitted by users is known to support External Entities, which can be abused to extract local files or trigger Server-Side-Request-Forgery (SSRF).
TB-03-021	Web Backend	It was discovered that the username was not properly checked in the backend during diagnostics upload. This allowed and attacker to upload diagnostic data for other users, and opened up the backend to further potential abuse via specially crafted usernames.
TB-03-033	Mac	It was found that the Mac client contained a configuration that made it ignore all TLS certificate warnings on certain domains. A malicious attacker, using only a self-signed TLS certificate, could leverage this weakness to intercept and modify traffic to these domains.
TB-04-001	Server Hardening	It was found that ASLR setting in *sysctl.conf* was not set properly, making exploits for buffer overflows or other memory corruption vulnerabilities easier to exploit.

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

Informational Severity (12)		
TB-03-004	Backend	It was found that the tested server had a lot of ssh-rsa keys stored in the */tmp* directory on the server. Some of the ssh-rsa keys contained host-names and email addresses belonging to TunnelBear employees.
TB-03-005	Backend	It was discovered that the test server was using an outdated version of BIND that had three publicly known vulnerabilities.
TB-03-010	Web	TunnelBear web application sent clear-text HTTP links on all its emails. This unnecessarily exposed the users of TunnelBear to attackers with the ability to modify network communications.
TB-04-007	Backend	It was discovered that RADIUS authentication script would always successfully authenticate certain hard-coded credentials.
TB-03-019	Web Backend	It was discovered that the web application deploys two catch-all rules in the route configuration. This allows it to call any controller and any action defined as public.
TB-03-030	Android	It was found that the TunnelBear Android application caches data in the SD Card. Unlike the app storage protected by Android, this location should be considered insecure.

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TB-03-031	Android	It was discovered that the Android app supports versions from API 14 (Ice Cream Sandwich) to 21 (Lollipop). This configuration can introduce security issues as API versions lower than 16 have world-wide-readable Android log files. This means that any application without any special permissions could read sensitive information saved by the TunnelBear app in these logs. It must be noted that no security or privacy related information of the TunnelBear app could be discovered in the logs during this assignment.
TB-03-032	Misc	A static username/password combination was found during source code audit. Similarly, audit identified that a password for client certificate included in the app bundle was embedded in source code.
TB-04-002	Web	It was found that the originally filed issue *TB-03-008 Web: Session cookie scoping allows phishing attacks (Low)* was not fully addressed yet.
TB-04-003	Web	It was found that security headers are not displayed on error pages, this might allow exploitation of Clickjacking or Copy&Paste XSS issues in various edge case scenarios

Dr.-Ing. Mario Heiderich, Cure53
Rudolf Reusch Str. 33
D 10367 Berlin
cure53.de · mario@cure53.de

TB-04-005	Web	The web application was found to expose URLs to non-production routes and views, including obsolete code. Some of that code was using an outdated jQuery library, known to have multiple vulnerabilities
TB-04-012	Android	Several recommendations regarding the choice of ciphers used to protect data stored at rest were made.