

Yealink Android Room System Security White Paper

Yealink Security Team



Yealink Network Technology Co., Ltd.

Copyrighted All Rights Reserved

Table of Contents

Overview	4
1 Hardware Interface Security	4
1.1 Debugging Interface	4
2 System Security	5
2.1 Partition Encryption.....	5
2.2 Secure Boot.....	5
2.3 Address Space Layout Randomization	5
2.4 External Storage.....	5
2.5 Firmware Security.....	5
2.6 System Security Testing	6
3 Coding Security	6
3.1 Universal Coding Security.....	6
3.2 App Coding Security	7
3.3 Codebase Management	8
3.4 Third-party Components/Libraries.....	8
3.5 Channel/Port Security	8
3.6 Coding Security Specifications	9
4 Application Security	9
4.1 Permission Security	9
4.2 Session Security.....	10
4.3 Anti-brute force cracking.....	10
4.4 Loss Prevention.....	10
4.5 Data Management	10
4.6 Loophole Attack Protection	10
5 Communications Security	11
5.1 Protocol/Service Security.....	11
5.2 Wireless Transmission Security	11
6 Data Security	11
6.1 Certificate Security.....	11
6.2 Encryption Algorithm.....	12
6.3 Hash Algorithm	12
6.4 Weak Algorithm Scanning.....	12
6.5 Password Security	12
6.6 Configuration Security	13
6.7 Diagnostic Security	13
6.8 URL Security	13

7	Security Testing and Release	14
7.1	Product Launch Process.....	14
7.2	Firmware Security Testing	14
7.3	Code Security Specifications	14
8	Privacy Security	15
9	Appendixes	16
	Appendix 1 Vulnerability Scanning Report	16
	Appendix 2 Weak Algorithm Scan Report	18
	Appendix 3 URL Description	19

Overview

This white paper aims to explain the security design, testing and improvements of the Android Room System of Yealink devices, including Yealink RoomCast, Yealink MeetingBar series. The Yealink security team attaches great importance to the security performance of devices. It regularly conducts internal tests and hires third-party security agencies to perform comprehensive security vulnerability scanning and penetration testing, and releases security performance improvements through hot-fix patches or version updates.

This white paper introduces the security performance from various aspects. The security test data cited include test reports of well-known security agencies in the industry and test results of authoritative security tools. Yealink will update the white paper from time to time, and the latest version of this white paper will be available on Yealink's official website.

1 Hardware Interface Security

1.1 Debugging Interface

- Disabling of the device's debugging interface by default

All hardware debugging interfaces are disabled by default after release from the factory to prevent unnecessary physical debugging interface exposure and information transmission.

- Removal of the debugging interface's silk screen

The silk screen has been removed from the PCB of all devices to prevent reverse engineering.

2 System Security

2.1 Partition Encryption

The sensitive information of the device has been encrypted and stored in a memory chip (flash, nand, emmc, etc.) through bit-level data encryption of the operating system partition, and the relevant information cannot be obtained by cracking

2.2 Secure Boot

The device supports secure boot. When booting, the firmware or Flash key partitions are legally loaded and verified to ensure that the system can start normally after successful security verification. It also prevents others from gaining control of the device through access to the SHELL interface by modifying boot parameters

2.3 Address Space Layout Randomization

Address space layout randomization (ASLR) protection is enabled to prevent buffer overflow.

2.4 External Storage

The device is prohibited from running programs or scripts from external storages (SD cards, USB flash drives, network storages, etc.). This prevents the system from being implanted with malware or scripts.

2.5 Firmware Security

The firmware data is encrypted with a high-strength encryption algorithm, and the firmware cannot be cracked or tampered with.

The firmware installation process has security checks that identifies incorrect firmware and tampered firmware and prevents their installation.

2.6 System Security Testing

Penetration tests are conducted to ensure that secure boot functionality, firmware analysis, malicious firmware updates, etc. are all safe.

Before the official release of the firmware, the Yealink security team also uses a variety of mainstream anti-virus software to scan the firmware to ensure that the firmware is not infected or embedded with viruses/Trojans. For details of the scan report, please refer to Appendix 1.

Yealink's security team and IT department regularly perform static and dynamic vulnerability scanning and penetration testing on the production environment and the company's internal network environment to ensure that the software development, firmware packaging and device production processes are completed in a secure network environment.

3 Coding Security

3.1 Universal Coding Security

- **Random number generation function**

Both device systems and applications use internal random algorithms to generate random numbers required for authentication or encryption to ensure security

- **String or memory manipulation functions**

The system and application use safe string or memory manipulation functions to prevent the use of strings or memory functions that do not perform bounds checking, which may be used by attackers for buffer overflow attacks

- **Format string function parameters**

The system and application use format string functions without external controllable variables as parameters to prevent serious harm caused by format string vulnerabilities.

3.2 App Coding Security

- **Stack cookie overflow protection**

Linux program code is used for compilation, and the CANARY stack overflow protection option of the Linux program is enabled by default

- **Stack non-executable protection**

Linux program code is used for compilation, and the NX stack non-executable protection option of the Linux program is enabled by default.

- **Base address random loading protection**

The PIE application base address random loading protection is enabled. Position-independent code / position-independent executable

- **Linux program code compilation**

When compiling Linux program code, the Strip function is used to delete the debug symbol table to increase the difficulty of reverse analysis and reduce the program size.

- **System call function parameters**

When using system call functions in coding, external controllable parameters should be filtered to prevent command injections

3.3 Codebase Management

Device-related code bases have strict permission management mechanisms and company red line requirements. In order to prevent source code leakage, uploading to public or semi-public services such as Github, Gitee and other public code repositories without permission is prohibited.

3.4 Third-party Components/Libraries

When third-party components/open source software are involved in a device, versions of prohibited categories are not used, and there are no residual, unused, or low-version third-party components. Externally disclosed vulnerabilities have been fixed in accordance with official patches, versions, or schemes

3.5 Channel/Port Security

- Management channel security

The terminal supports access protection to the interface of the management plane, meaning that it is not allowed to directly log in and connect to the management interface from the user plane. Different users can only view different information and manage different contents.

- Port attack defense

All external communication connections of the terminal must be indispensable for the operation and maintenance of the terminal (functions that customers need to use). The communication port used is described in the product communication matrix document, and the dynamic listening port is limited to a certain reasonable range. A port scan tool is used for verification. Ports that are not listed in the communication matrix must be closed (NMAP port scan; refer to Appendix 2 for the scan report).

To strengthen the defense against DDos attacks, open ports are limited, and applications using open ports have the ability to detect

and filter malformed packets and packets that do not meet the rules.

- Access security

All physical interfaces, communication ports, and protocols that can manage terminals have access authentication mechanisms. The login service provided by the terminal should require the user to re-authenticate if the user has not carried out any operations for a period of time.

3.6 Coding Security Specifications

In addition to the above-mentioned security design, Yealink has strict internal coding security specifications for the R&D team. During every code update, the code is reviewed and a reliability verification is carried out.

4 Application Security

4.1 Permission Security

- Authentication bypass

Each request that requires authorized access must verify whether the user's session ID is legal and whether the user is authorized to perform this operation, so as to prevent unauthorized access and ensure that the user cannot directly access resources that require authorization authentication

- Vertical ultra vires

Low-privilege users cannot access resources that can only be accessed by high-privilege users through vertical unauthorized access which bypasses authentication and authorization

- Server authentication

Terminal devices involving user authentication are currently operating on the server side to avoid the risk of being hijacked to bypass authentication

4.2 Session Security

Yealink's internal session security management mechanism is adopted, and session security is also ensured through various methods, such as high-quality pseudo-random number generator, high-strength encryption function, etc.

4.3 Anti-brute force cracking

The authentication module adopts an anti-brute force cracking mechanism. After the verification code fails or multiple consecutive login attempts fail, the account or IP is locked, and needs to be unlocked to continue access.

4.4 Loss Prevention

When the device transmits the user name and password to the server, a security protocol must be used. For example, HTTPS must be used for WEB login, and the password must not appear as plaintext in the configuration file, log, cookie, and debugging trace information.

4.5 Data Management

The processing of data by the device is done on the device side. In other words, the final input processing (authentication), validity (data legitimacy), etc. are all performed on the terminal.

4.6 Loophole Attack Protection

For terminal equipment, Nessus, Acunetix, and other common terminal vulnerability scanning tools used in the industry are used to ensure that the equipment is free of medium and high risk vulnerabilities. Please refer to Appendix 3 for the test report

5 Communications Security

5.1 Protocol/Service Security

- TLS 1.2 support

The device encrypts communications using an encrypted transport protocol, with additional encryption for sensitive information while the data is in transit. This avoids the risk of information leakage or tampering due to the use of plaintext transmission protocols such as HTTP

- HTTPS support

The device supports HTTPS security protocols. For example, web pages, AutoP update, and other functions support HTTPS, which is more secure.

5.2 Wireless Transmission Security

- 802.1x support

802.1X is supported, and the certificate and TLS key will be burned in via the USB cable during pairing. It will not be delivered in the air and therefore has no possibility of being stolen.

6 Data Security

6.1 Certificate Security

Each device is pre-configured with its own unique device certificate at the factory, which is encrypted using the SHA256 algorithm. The preset device certificate is used by default during mutual authentication with the server.

6.2 Encryption Algorithm

The device uses secure encryption algorithms, and the key length meets the minimum requirements of the corresponding algorithm. For example, no less than 128 bits for AES, no less than 256 bits for SHA, etc. Insecure encryption algorithms, such as DES, TDES, and RC4 are not used.

6.3 Hash Algorithm

We use secure hash functions with no less than 256 bits, such as SHA256, and do not use hash functions with existing security risks, such as MD5 or SHA1.

6.4 Weak Algorithm Scanning

The Zenmap scanning tool is used for weak algorithm scanning. Algorithm protocols that do not meet security requirements are not used. Refer to Appendix 4 for details.

6.5 Password Security

- Password usage

Users can see obvious prompts when using the default password. Security requirements must be met for password modifications to take effect. Anti-brute force cracking is supported. RSA encryption for password transmission is adopted.

- Password usage

The password of any input box cannot be displayed in plaintext, nor does it support copying. The account will be automatically logged out when it times out. Low-privilege users cannot modify the information of high-privilege users. The exported configuration files and logs do not contain passwords so as to ensure security.

- Password Protection
 1. Passwords are not allowed to be stored in the phone in plaintext or using BASE64 encoding.
 2. When entering a password, it cannot be displayed in plaintext, nor be copied.
 3. In scenarios where password recovery is not required, an irreversible algorithm must be used for encryption.
 4. The password cannot be stored in plaintext inside the system.

6.6 Configuration Security

When exporting the user configuration, the password cannot be exported. Passwords in logs cannot be displayed in plaintext. Configuration files can be encrypted with the AES256 high-strength algorithm. Exported configuration files, such as config, cannot be directly decompressed and must be decrypted by a special security decryption tool before decompression.

6.7 Diagnostic Security

When exporting the user configuration, the password cannot be exported. Passwords in logs cannot be displayed in plaintext. Exported config.bin files cannot be directly decompressed and must be decrypted by a special security decryption tool before decompression.

6.8 URL Security

It is currently prohibited for the product software of devices to include unpublished public network addresses (including public network IP addresses, public network URL addresses/domain names, and email addresses) which are invisible on the user interface or not described in the product documentation. The URLs currently involved are all relevant Support websites. Refer to Appendix 5 for details.

7 Security Testing and Release

7.1 Product Launch Process

Yealink follows the Secure Software Development Life Cycle (S-SDLC), emphasizing security in all stages of product requirements, design solutions, and coding specifications. Before any software version is released, it must go through an Alpha version test, a Beta version test, and a UAT test. At each stage, the security team participates in the penetration test of software and hardware, attack surface analysis, and security scanning.

7.2 Firmware Security Testing

Yealink's security team and IT department regularly perform static and dynamic vulnerability scanning and penetration testing on the production environment and the company's internal network environment to ensure that the software development, firmware packaging and device production processes are completed in a secure network environment.

Before the official release of the firmware, the Yealink security team also uses a variety of mainstream anti-virus software to scan the firmware to ensure that the firmware is not infected or embedded with viruses/Trojans.

Penetration tests are conducted to ensure that secure boot functionality, malicious firmware analysis, tampered firmware updates, etc. are all safe.

7.3 Code Security Specifications

Yealink has strict internal coding security specifications for the R&D team. During every code update, the code is reviewed and a reliability verification is carried out.

Device-related code bases have strict permission management mechanisms and company red line requirements. In order to prevent source code leakage, uploading to public or semi-public services such as Github, Gitee and other public code repositories without permission is prohibited.

8 Privacy Security

Throughout the life cycle of the phone (starting from release from the factory), all user data, including account information, contacts, call records, audio and video call information, etc., are managed directly and locally by the customer in the customer device end. Yealink follows the local privacy protection policy and does not illegally acquire, store, or collect unauthorized customer data by any means.

The user data of the phone device is stored in an encrypted form to reduce the risk of privacy leakage caused by improper user operations.

The network connection and network access generated by the phone device are determined by the company's internal network and user behavior operations. The phone device does not initiate any communication with any IP address or server without the customer's permission.

If the user's device fails and needs to be diagnosed and repaired, Yealink has no way to obtain the current information of the device and therefore requires the cooperation and authorization of the customer to export the phone's diagnostic file locally for troubleshooting.

9 Appendixes

Appendix 1 Vulnerability Scanning Report

Nessus scan report:



Report generated by Nessus™

AX0 V22 SP

Thu, 16 Jun 2022 14:16:28 China Standard Time

TABLE OF CONTENTS

Vulnerabilities by Host

- 10.50.70.61

Vulnerabilities by Host

[Collapse All](#) | [Expand All](#)

10.50.70.61



Severity	CVSS v2.0	Plugin	Name
MEDIUM	6.4	51192	SSL Certificate Cannot Be Trusted
INFO	N/A	10863	SSL Certificate Information
INFO	N/A	70544	SSL Cipher Block Chaining Cipher Suites Supported
INFO	N/A	21643	SSL Cipher Suites Supported
INFO	N/A	57041	SSL Perfect Forward Secrecy Cipher Suites Supported

Hide

Specifically, the reason for the middle-level vulnerability, "SSL Certificate Cannot Be Trusted", is that the device certificate is issued by Yealink and the Yealink root certificate is not installed on the server side. It has nothing to do with the client and has no security risk.

Note: Nessus, a proprietary security scanning tool developed by Tenable, is an international mainstream system vulnerability scanning and analysis software. Its vulnerability database is updated in real time, which provides high-speed and accurate scanning with minimal false positives.

Acunetix scan report:

Scan of 10.50.70.61

Scan details

Scan information	
Start time	14/06/2022, 11:23:07
Start url	https://10.50.70.61/api#/status-general
Host	10.50.70.61
Scan time	13 minutes, 28 seconds
Profile	Full Scan
Server information	nginx/1.20.2
Responsive	True
Server OS	Unknown

Threat level

Acunetix Threat Level 1

One or more low-severity type vulnerabilities have been discovered by the scanner.

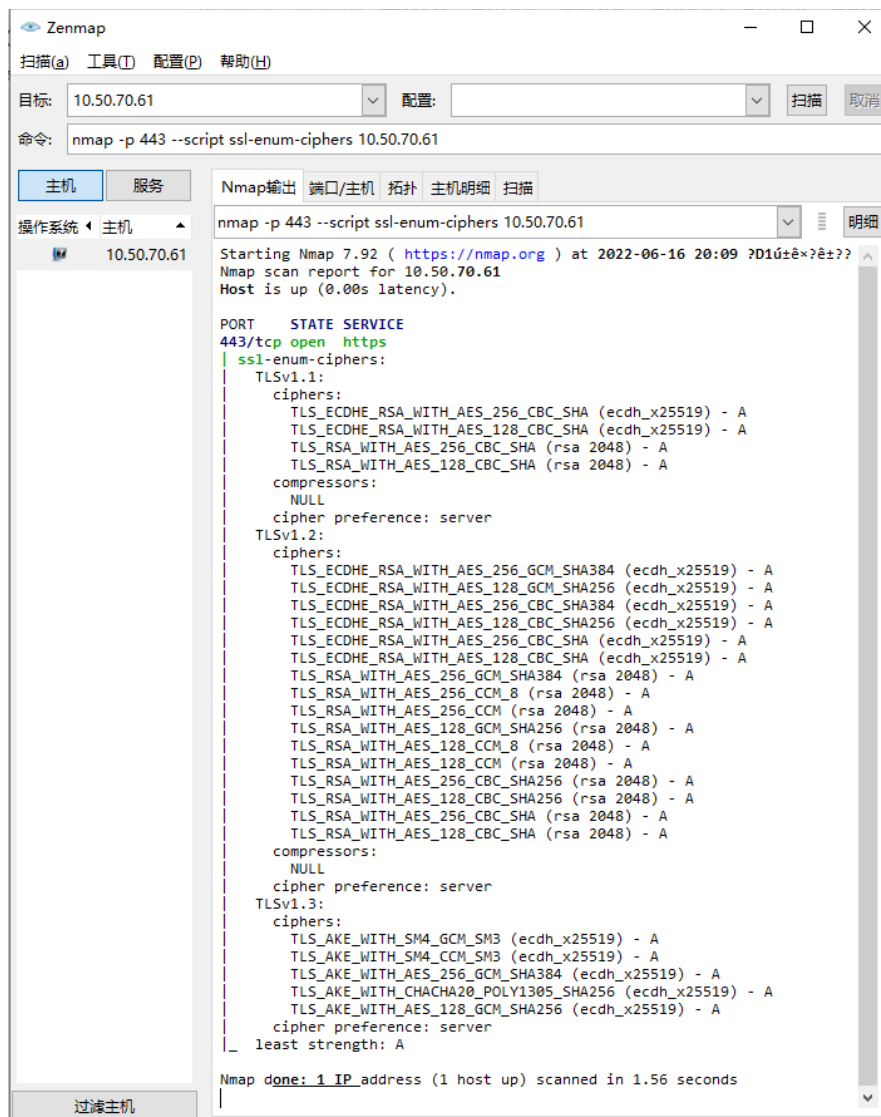
Alerts distribution

Total alerts found	3
High	0
Medium	0
Low	1
Informational	2

Result: According to the Acunetix scan, there are no medium or high level vulnerabilities

Note: Acunetix, the world's leading network application security scanning software, is mainly used for network application related security scanning

Appendix 2 Weak Algorithm Scan Report



```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-16 20:09 ?D1ú±è×?ê±??
Nmap scan report for 10.50.70.61
Host is up (0.00s latency).

PORT      STATE SERVICE
443/tcp   open  https
| ssl-enum-ciphers:
| TLSv1.1:
|   ciphers:
|     TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (ecdh_x25519) - A
|     TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A
|     TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
|     TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
|   compressors:
|     NULL
|   cipher preference: server
| TLSv1.2:
|   ciphers:
|     TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
|     TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
|     TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (ecdh_x25519) - A
|     TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (ecdh_x25519) - A
|     TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (ecdh_x25519) - A
|     TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A
|     TLS_RSA_WITH_AES_256_GCM_SHA384 (rsa 2048) - A
|     TLS_RSA_WITH_AES_256_CCM_8 (rsa 2048) - A
|     TLS_RSA_WITH_AES_256_CCM (rsa 2048) - A
|     TLS_RSA_WITH_AES_128_GCM_SHA256 (rsa 2048) - A
|     TLS_RSA_WITH_AES_128_CCM_8 (rsa 2048) - A
|     TLS_RSA_WITH_AES_128_CCM (rsa 2048) - A
|     TLS_RSA_WITH_AES_256_CBC_SHA256 (rsa 2048) - A
|     TLS_RSA_WITH_AES_128_CBC_SHA256 (rsa 2048) - A
|     TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
|     TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
|   compressors:
|     NULL
|   cipher preference: server
| TLSv1.3:
|   ciphers:
|     TLS_AKE_WITH_SM4_GCM_SM3 (ecdh_x25519) - A
|     TLS_AKE_WITH_SM4_CCM_SM3 (ecdh_x25519) - A
|     TLS_AKE_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
|     TLS_AKE_WITH_CHACHA20_POLY1305_SHA256 (ecdh_x25519) - A
|     TLS_AKE_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
|   cipher preference: server
|_ least strength: A

Nmap done: 1 IP address (1 host up) scanned in 1.56 seconds
```

Info: The results obtained by the algorithm scanning tool show that the algorithms used by the terminal are all of the highest security level A, and weak algorithms with potential security risks are not used.

Appendix 3 URL Description

It is currently prohibited for the product software of devices to include unpublished public network addresses (including public network IP addresses, public network URL addresses/domain names, and email addresses) which are invisible on the user interface or not described in the product documentation. The URLs currently involved are all relevant Support websites and server address. Refer to the following table:

Module	URL	Description	Ports
Yealink Zero Touch Deployment	https://rpscloud.yealink.com https://rps.yealink.com	Yealink RPS Service(Redirection and Provisioning Service) for zero touch deployment. This service will be triggered only on the first boot, or until you factory reset the device. If you don't need zero touch deployment service you could block this URL. Note: RPS is a free service for Yealink customers all over the world. By simply entering the MAC address of Yealink device and the URL of the provision server into Yealink's RPS, upon initial boot-up the device can be redirected to its pre-assigned server for configuration updating. Consequently, it is no longer necessary to unpack, configure or repack a phone before shipping to the customer. This greatly simplifies mass deployment by automatically configuration, and saves a lot of time and money for deployment.	tcp 443
Internet Connectivity Detection	http://www.google.com http://www.msftncsi.com http://g.cn http://www.msftncsi.com	To check Internet connectivity	tcp 80
Yealink Device management	dmtp.yealink.com dm.yealink.com rps.yealink.com rpscloud.yealink.com	Yealink Device Management Service Global Server Location: Germany/United States If you don't need to use Yealink device management	tcp 443
Network Time Service	cn.pool.ntp.org north-america.pool.ntp.org pool.ntp.org	The Network Time Protocol (NTP) is a networking protocol for clock synchronization between computer systems over packet-switched, variable-latency data networks. We use following global NTP server as default: cn.pool.ntp.org north-america.pool.ntp.org pool.ntp.org	udp 123

Disclaimer

This white paper is provided for informational purposes only and does not convey any legal rights to any intellectual property in any Yealink product. You may copy and use this paper for your internal reference purposes only.

YEALINK MAKES NO WARRANTIES, EXPRESS OR IMPLIED OR STATUTORY AS TO THE INFORMATION IN THIS WHITE PAPER.

To learn more about Yealink IP Phone Series devices, visit [our website](#)

Copyright: Yealink Network Technology Co., Ltd.