ITUPublications
Recommendations

International Telecommunication Union
Standardization Sector

Recommendation

ITU-T X.1282 (11/2023)

SERIES X: Data networks, open system communications and security

Cyberspace security – Available

Security measures for countering password-related online attacks

ITU-T X-SERIES RECOMMENDATIONS

Data networks, open system communications and security

PUBLIC DATA NETWORKS	X.1-X.199
OPEN SYSTEMS INTERCONNECTION	X.200-X.299
INTERWORKING BETWEEN NETWORKS	X.300-X.399
MESSAGE HANDLING SYSTEMS	X.400-X.499
DIRECTORY	X.500-X.599
OSI NETWORKING AND SYSTEM ASPECTS	X.600-X.699
OSI MANAGEMENT	X.700-X.799
SECURITY	X.800-X.849
OSI APPLICATIONS	X.850-X.899
OPEN DISTRIBUTED PROCESSING	X.900-X.999
INFORMATION AND NETWORK SECURITY	X.1000-X.1099
SECURE APPLICATIONS AND SERVICES (1)	X.1100-X.1199
CYBERSPACE SECURITY	X.1200-X.1299
Cybersecurity	X.1200-X.1229
Countering spam	X.1230-X.1249
Identity management	X.1250-X.1279
SECURE APPLICATIONS AND SERVICES (2)	X.1300-X.1499
CYBERSECURITY INFORMATION EXCHANGE	X.1500-X.1599
CLOUD COMPUTING SECURITY	X.1600-X.1699
QUANTUM COMMUNICATION	X.1700-X.1729
DATA SECURITY	X.1750-X.1799
IMT-2020 SECURITY	X.1800-X.1819

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T X.1282

Security measures for countering password-related online attacks

Summary

Recommendation ITU-T X.1282 analyses security risks of password-related online attacks in service systems and provides security measures to mitigate security threats and challenges.

Based on features of password-related online attacks, security measures include the completely automated public Turing test to tell computers and humans apart (CAPTCHA), multi-factor certification, session control, log audit, security design of registration interface, security design of retrieving password interface, security design of login interface, security policy of login password, anomaly pattern analysis, data analysis, policy optimization, hierarchical services, risk early warning, user reminders and other related technical requirements.

Recommendation ITU-T X.1282 provides security risks analysis and security considerations to mitigate password-related security risks into each phase of the service life cycle, thus advancing the business application and security requirements together to ensure a balanced approach during the life cycle of service systems. It provides a baseline to all service systems that use password login mechanisms, and additional filters for critical applications.

History *

Edition	Recommendation	Approval	Study Group	Unique ID
1.0	ITU-T X.1282	2023-11-13	17	11.1002/1000/15713

Keywords

Password-related online attacks, security threats, security measures.

* To access the Recommendation, type the URL https://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents/software copyrights, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the appropriate ITU-T databases available via the ITU-T website at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T X.1282

Security measures for countering password-related online attacks

1 Scope

This Recommendation analyses the security risks of password-related online attacks in service systems and provides security measures based on hypertext transfer protocol (HTTP) header enrichment technology that could mitigate the security threats and challenges.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

None.

3 Definitions

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

APP	Application
APPID	Application Identity
APPKEY	Application Key
CAPTCHA	Completely Automated Public Turing Test to tell Computers and Humans Apart
CSP	Credential Service Provider
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ID	Identifier
IMT-2020	International Mobile Telecommunication-2020
IP	Internet Protocol
JSSDK	JavaScript Software Development Kit
MSISDN	Mobile Station Integrated Services Digital Network
PDN	Public Data Network
PGW	Public data network Gateway
SDK	Software Development Kit
SIM	Subscriber Identity Model
SMS	Short Message Service

UPF User Plane Function

URL Uniform Resource Locator

5 Conventions

In this Recommendation:

The phrase "is required to" indicates a requirement that is required to be strictly followed and from which no deviation is permitted if conformance to this Recommendation is to be claimed.

The phrase "is recommended" indicates a requirement that is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

The phrases "can optionally" and "may" indicate an optional requirement that is permissible, without implying any sense of being recommended. These terms are not intended to imply that the vendor's implementation is required to provide the option and the feature can be optionally enabled by the network operator or service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with this Recommendation.

6 Overview

User authentication process is the most critical part of access control model. The traditional user identification process generally uses a secondary authentication process such as account password plus a short message service (SMS) verification code to ensure security. However, the interaction process of account password authentication is cumbersome and requires users to remember passwords with complex rules. Furthermore, the attackers may obtain the user's login authentication identity using a credential stuffing attack or SMS sniffing attack.

Therefore, traditional identity authentication methods are not ideal access methods for mobile applications (APPs) in terms of convenience and security.

[b-GSMA PDATA.03] introduces seamless authenticator implementation approaches by HTTP header enrichment technology. The technology, based on the enhanced function of the core network gateway device (gateway general packet radio service support node/public data network gateway (PGW)/user plane function (UPF)) header, identifies the user international mobile station integrated services digital network (MSISDN) number by parsing the upstream traffic data packets, and authenticates user identity based on the MSISDN.

6.1 HTTP header enrichment technology

HTTP header enrichment technology of the core network gateway adds specific information to the HTTP header relating to some specific uniform resource locator (URL).

The HTTP header enrichment process for PGW is shown in Figure 1.

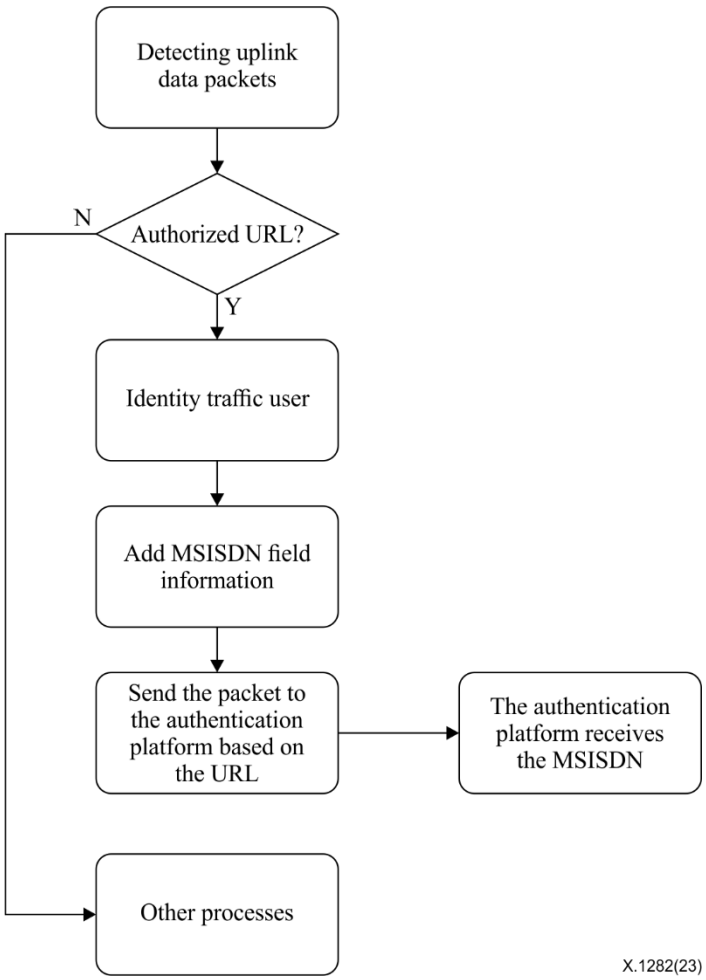

Figure 1 – HTTP header enrichment process for PGW

The core network gateway can process HTTP packets based on URL or Internet protocol (IP) address detection. From the gateway's inner database, the associated user information is hosted according to the user-bound public data network (PDN), and the corresponding MSISDN field is inserted in the HTTP protocol request header.

The core network gateway forwards the upstream packet to the authentication platform, which resolves the user's MSISDN field from the HTTP packet to identify the user identity of the current upstream traffic packet subject.

6.2 Security risks

In traditional authentication services, there are three main types of security risk in the authentication process, namely: authentication information leakage; server session hijacking; and credential service provider (CSP) spoofing attacks.

– Risk of authentication information leakage means that after an attacker obtains user authentication information through brute force, dictionary, replay, etc., the authentication information is successfully authenticated on the server side.

– Session hijacking means that the attacker can access the service system without authorization if the communication process between the authentication service system and the client is hijacked.

– A CSP spoofing attack means that the attacker bypasses the security policy of the service system by constructing attack data to achieve unauthorized access.

Although the security risks described in the previous paragraph differ from attack methods, the main goals of both are unauthorized access to the system, which means traditional authentication services, to ensure the security of the system, need to prevent attackers from using different attack methods to implement unauthorized access attacks.

6.3 Security authentication process based on HTTP header enrichment technology

The identity authentication link is based on the HTTP header enrichment technology of the operator's core network gateway to achieve user identity authentication. The authentication process should follow the relevant processes defined in the oAuth/OAUTH 2.0 standard [b-IETF RFC 6749].

The authentication process based on mobile user numbers consists of three main roles:

a) service provider: the entity that provides services to the end user;

b) resource owner: owns the data in the resource server;

c) authentication platform: the service provider platform based on the identity authentication of the user number provides identity authentication, user authorization and platform authentication services.

Figure 2 shows the basic flow of a security identity authentication process based on HTTP header enrichment technology.

1) End user authentication request

   When the user is using the service from the service provider, in the scenario where the user identity needs to be authenticated. The client initiates an HTTP request. When the request passes through the core network gateway, the gateway's header enhancement technology inserts the user's MSISDN into the HTTP message header, then sends the message to the authentication platform.

2) Authorization confirmation

   After the authentication platform completes the authentication of the user identity, the AuthZ (authorization) code is returned from the authentication platform. When the user grants the authorization, the authentication platform returns the identifier (ID) document token to the client.

3) Request user identity

   The service provider carries the ID document token to the authentication platform to confirm the user authentication result.

4) Access user identity

   The user identity (MSISDN) is returned after the ID document token is successfully verified.

Security authentication based on HTTP header enrichment technology provides a new security method to mitigate the authentication information leakage, session hijacking, CSP spoofing attack.

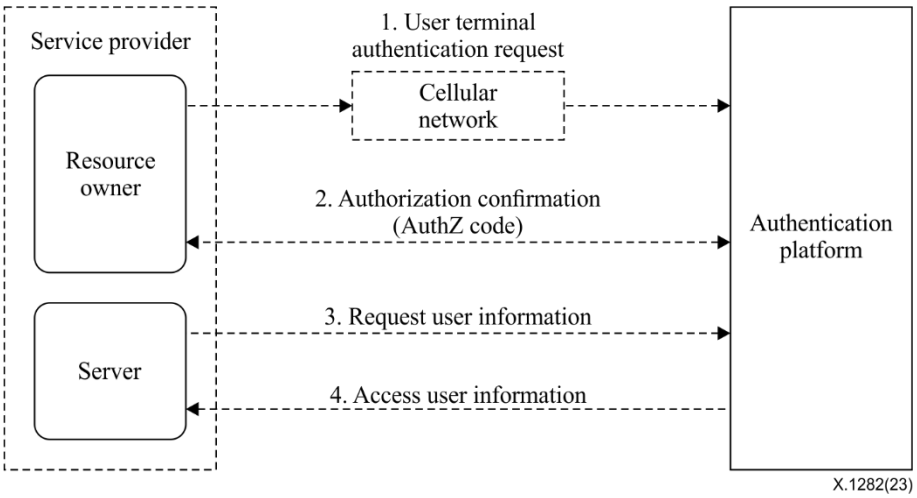

Figure 2 – Basic flow of identity authentication process

A user can log in to an APP or access the service system without entering any account name or password with the help of security authentication based on HTTP header enrichment technology. User authentication information is not exposed when transmitted and does not need to be stored in the server database. This authentication method can effectively avoid user authentication information leakage.

7 Security threats in the authentication process

According to the associated party analysis model of password-related online attacks on a service system proposed in clause 6, the security risks of the authentication process in a service system are divided into three source categories: authenticator compromise; transaction compromise; and verifier impersonation.

7.1 Authenticator compromise risks

The username, password and other authentication information of the service system are obtained by the attacker, who uses them to successfully authenticate and gain unauthorized access to the service system. Methods include brute force attack, replay attacks, phishing and account credential enumeration attack.

– Brute force attack includes two kinds: by key guessing and enumeration. A key-guessing attack means that the attackers guess the key and decrypt the code according to the intercepted cipher text. An enumeration attack substitutes the elements of a password one by one until the real password is found. In practical application, types of enumeration attack are account name exhaustion and password exhaustion.

– Replay attack: after obtaining a successfully authenticated packet, an attacker repeatedly uses it to log in to and gain unauthorized access to the service system.

– Phishing: by pretending to be a trusted individual or enterprise, the attacker entices users to click into a website to enter sensitive personal information, such as username, password and credit card details. Phishing attacks are usually carried out by email or instant communication (chat APPs, SMS, etc.).

– Account credential enumeration attack: the attacker collects username and password information leaked from service system A and generates the corresponding dictionary table, attempts to log in to service system B and other service systems in batch, and obtains a series of usernames and passwords that can be used to log in.

7.2 Transaction compromise risks

The attacker accesses data in transmission in the service system to obtain unauthorized access to it. Attack methods include man-in-the-middle attack and session hijacking.

– Man-in-the-middle attack: the attacker is placed between the two sides of communication, usually between the client and server sides of a service system. After destroying the original communication line, the attacker intercepts messages from one side and forwards them (sometimes changed) to the other.

– Session hijacking: attackers use brute force cracking, network sniffing tools and other methods to obtain the session ID document of users, and then log in to obtain unauthorized access to the service system.

7.3 Verifier impersonation risks

The attacker cheats or tampers with the verification result of CSP to obtain unauthorized access to the service system.

– Spoofing CSP: the attacker uses defects in a CSP to construct false login information, spoofing CSP to obtain unauthorized access to the service system. An example is structured query language injection.

– Tampering with CSP verification results: attackers use technical means to intercept and modify CSP verification results, return them after tampering to the service system, and obtain unauthorized access to the service system.

8 Security authentication process via HTTP header enrichment technology

Security authentication process based on HTTP header enrichment technology includes processes for authentication, user authorization and platform verification.

The following roles are involved in this process.

a) APP: the business party that initiated the user identity authentication process.

b) Software development kit (SDK): the third party SDK provided for an APP. After integrating the toolkit, the APP can authenticate identity by calling the interfaces and methods provided in the toolkit.

c) PGW/UPF: a PGW provides functions such as user session management and bearer control, data forwarding, IP address allocation, and non-3GPP user access (a UPF network element is under the IMT-2020 network). In the authentication process, the role of PGW/UPF is to identify and insert user identity information into the message.

d) Authentication platform: the service provider platform based on identity authentication of the user number additionally provides user authorization and platform authentication services. Its main function is to issue and verify token information.

e) APP server: the server of the business party needs to carry a token to the authentication platform in exchange for the user's MSISDN in the platform verification process.

f) Resource owner: the owner who owns the data in the resource server.

8.1 Authentication process

See Figure 3.

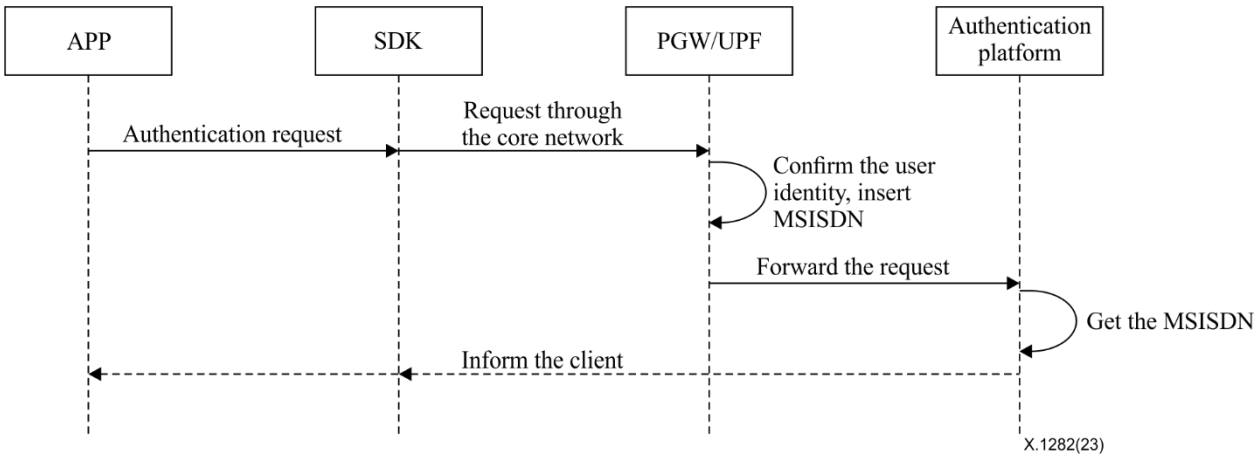

Figure 3 – Authentication process flow

The authentication process includes the interaction between the APP, SDK, PGW/UPF and authentication platform.

– Step 1: APP request for authentication.

– Step 2: APP request reaches the PGW/UPF through the base station.

– Step 3: PGW/UPF can process HTTP packets based on URL or IP address detection. From the gateway's inner database, the associated user information is hosted according to the user-bound PDN, and the corresponding MSISDN field is inserted in the HTTP protocol request header.

– Step 4: PGW sends the packet to the authentication platform based on the URL.

– Step 5: authentication platform gets the user's MSISDN from the HTTP packet to identify the user.

– Step 6: the authentication platform returns a flag to notify the SDK whether the authentication process has been successful.

8.2 User authorization process

See Figure 4.

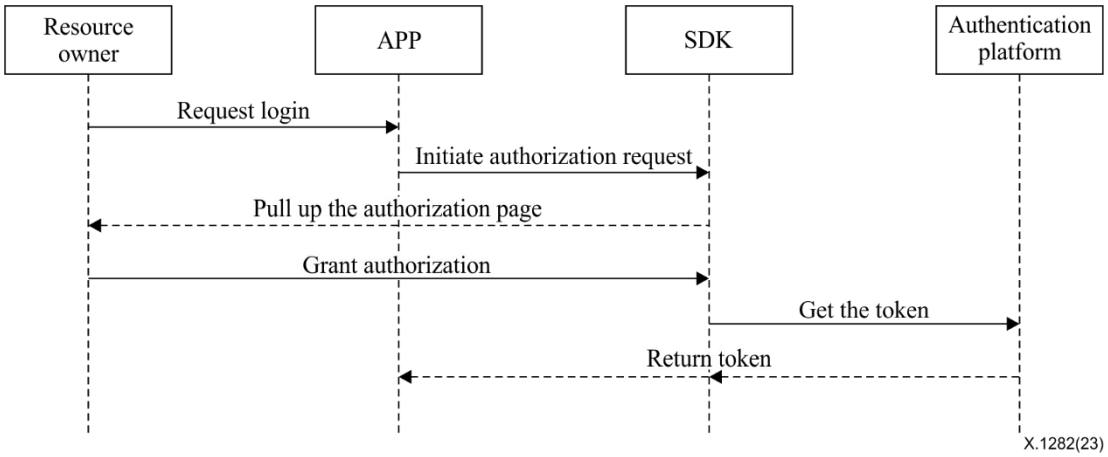

Figure 4 – User authorization process flow

The user authorization process involves the interaction between the resource owner, APP, SDK and authentication platform.

– Step 1: resource owner requests to log in to the APP.
– Step 2: APP sends an authorization request to the SDK.
– Step 3: SDK pulls up the authorization page.
– Step 4: resource owner needs to confirm on the authorization page to authorize the APP to obtain the user MSISDN.
– Step 5: SDK issues the request to get the authorization token.
– Step 6: the authentication platform returns the authorization token information to the SDK.

8.3 Platform verification process

See Figure 5.

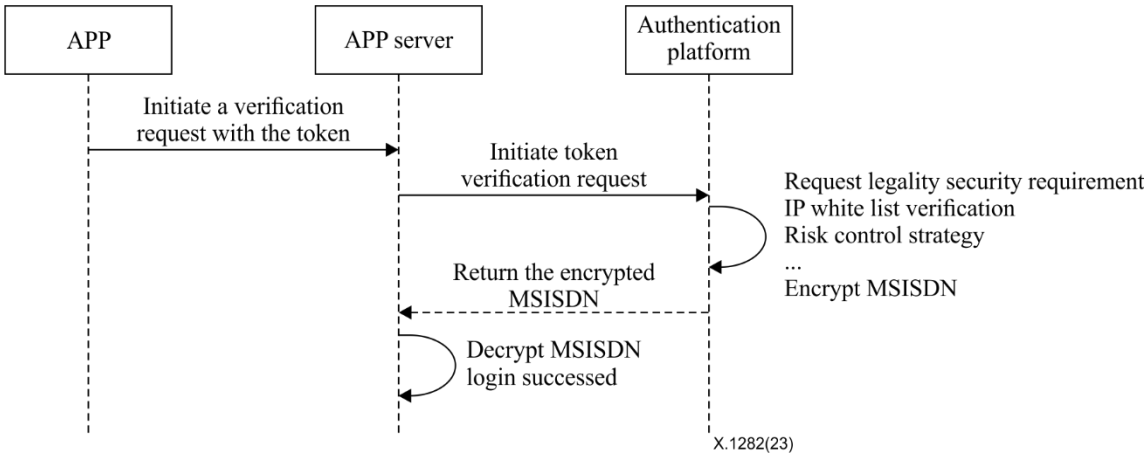

Figure 5 – Platform verification process flow

The platform verification process includes the interaction between the APP, APP server and authentication platform.

– Step 1: APP carries the authorization token obtained by the authorization process to the APP server.
– Step 2: APP server sends a token verification request to the authentication platform.
– Step 3: authentication platform verifies the request (request signature etc.), verifies the APP source IP, and evaluates the risk level.
– Step 4: authentication platform encrypts the MSISDN information with the private key of the business party and returns it to the APP server.
– Step 5: APP server decrypts the MSISDN information using the public key of the authentication platform to obtain the user MSISDN.

9 Client security

Identity authentication based on HTTP header enrichment technology requires mobile terminals working in cellular data network mode with a subscriber identity model (SIM) card. The service provider should initiate the client-side authentication request in the APP through the SDK or JavaScript software development kit (JSSDK). Client security requirements are needed to ensure that the authentication process is sufficiently secure.

9.1 APP security

Before the service provider APP accesses the SDK, it shall register and apply for service on the open platform. At the same time, the APP shall fill in information such as the identity of the APP on the open platform to verify the permission of the APP or page.

After completing the APP, the open platform will return the application identity (APPID) and application key (APPKEY) information to the service provider, who will use the corresponding APPID and APPKEY to call related services.

9.2 SDK security

9.2.1 SDK communication request verification

Verification of the request is required during communication between the SDK and authentication platform. Signature fields should be carried in the request parameters. The authentication platform should verify the validity of the signature before subsequent processes can be executed to ensure that all requests from the SDK are authorized.

9.2.2 SDK request message encryption protection

To ensure the security of the message information generated by the client, the important and sensitive fields should be encrypted before transmission.

9.2.3 HTTPS protocol for SDK interface

All requests initiated from the SDK side should comply with hypertext transfer protocol secure (HTTPS) to ensure that the request is transmitted in an encrypted channel.

9.2.4 Local data storage security

For data generated during the interaction between the SDK and authentication platform, if required to be stored locally in the SDK, the SDK should guarantee the uniqueness of the readable data. In the mobile terminal environment, inability to read or tamper with application data by other third party applications should be ensured.

9.2.5 SDK code obfuscation

SDK code should be obfuscated to prevent attackers from reverse engineering.

9.2.6 User privacy data security protection

User personal information that is not necessary for the authentication service should not be collected by the SDK. If it is necessary to collect the user information, it should be collected as infrequently as possible and with user consent.

9.2.7 SDK authorization page

The SDK should pull up the authorization page to ask for user confirmation to get their mobile phone number. After confirmation, the authorization token associated with the user mobile phone number information can be obtained. The authorization page should contain the following elements.

a) The page should contain words such as login or registration that are sufficient to describe the use of the current page.

b) The authorization page should prominently display an authorization confirmation button.

c) The authorization page should be able to obtain an authorization token only after the user clicks the authorization confirmation button.

d) The authorization page should prominently state that the purpose of the current authorization is to obtain the user local phone number.

e) The authorization page should contain the relevant agreement terms of service.

9.3 H5 security

9.3.1 H5 JSSDK code obfuscation

JSSDK code needs to be obfuscated to ensure that the readability of front-end code is reduced and the code security protection is strengthened.

9.3.2 H5 page reference verification

The service provider reports the page referrer address that calls the JSSDK code to the open platform. When the H5 page calls the JSSDK code, the JSSDK should collect the referrer address of the page and transmit the referrer field to the authentication platform. The authentication platform verifies the collected referrer information to verify its validity. Use of the JSSDK to access H5 services that have not been registered on the open platform is prohibited.

9.3.3 JSSDK communication request verification

When JSSDK communicates with the authentication platform, the request should be verified. The request parameters should carry the relevant signature parameters. The server side should verify the validity of the signature before allowing the execution of subsequent business logic to ensure that all requests from the JSSDK are authorized.

9.3.4 HTTPS protocol

The business request initiated from JSSDK should comply with HTTPS protocol to ensure that the request is transmitted in the encrypted channel.

9.3.5 Browser fingerprint

When JSSDK authenticates and authorizes users on the H5 page, it needs to collect the fingerprint information of the browser. The browser fingerprint information is used to associate and bind the process nodes of the primary identity authentication service to ensure process uniformity.

9.3.6 JSSDK authorization page

The JSSDK should pull up the authorization page to ask for user confirmation to get their mobile phone number. After confirmation, the authorization token associated with the user mobile phone number information can be obtained. The authorization page should contain the following elements.

a) The page should contain words such as login or registration that are sufficient to describe the use of the current page.

b) The authorization page should prominently display an authorization confirmation button.

c) The authorization page should be able to obtain an authorization token only after the user clicks the authorization confirmation button.

d) The authorization page should prominently state that the purpose of the current authorization is to obtain the user local phone number.

e) The page should contain the relevant agreement service terms.

f) The page should be protected from malicious coverage, and it is prohibited to obtain the SIM number without the user's permission or authorization without the user's knowledge.

9.3.7 User privacy data security protection

User personal information that is not necessary for service functions should not be collected by the JSSDK. If it is necessary to collect user information, it should be collected as infrequently as possible and with user consent.

10 Authentication platform security

The authentication platform provides standard identity authentication services for service providers through SDK or JSSDK. In the process of service provision, it is required to ensure the validity of access to clients and service provider platforms, as well as the security of data transmission.

10.1 Request verification

10.1.1 Client verification

a) Request signature verification: For all requests from clients, the signature should be verified to guarantee whether the service provider that initiates the request is authorized.

b) Business verification: For all requests from clients, the APPID of the request initiator should be verified to ensure that the service provider is registered on the open platform.

c) APP verification: For APPs, the package name, package signature (Android), and BundleID (iOS) should be verified to ensure the validity of the APP identity. For H5 applications, the referrer address of H5 pages should be verified to ensure the validity of their source.

d) Source IP verification: the authentication platform shall verify the source IP for the number acquisition request of the client and intercept the identity authentication request whose source is an unauthorized IP of the gateway.

10.1.2 Service provider verification

IP whitelist verification: for business requests from service providers, whether the source IP address of the service provider is consistent with the IP address registered on the open platform should be verified to ensure the validity of the source of the request platform.

10.2 Data encryption and decryption security

The client needs to encrypt and protect key fields related to the business, and the authentication platform should use the corresponding private key to decrypt the fields before executing the subsequent business logic.

10.2.1 Client data encryption and decryption security

The client needs to encrypt and protect key fields of the requests that are initiated by the APP, and the authentication platform should decrypt the fields before executing the subsequent business logic.

10.2.2 Service provider data encryption and decryption security

a) Important request and response fields should also be encrypted to guarantee the security of communication between the authentication and service provider platforms. The encryption keys shall be properly and safely maintained on both platforms.

b) The response data from the authentication platform should include a signature to prevent the result from being tampered with by a third party.

10.3 User data security management

The authentication platform needs to encrypt transmission, encrypt storage, desensitize display and process other user data as required.

10.4 Business risk control security

The authentication platform needs to put forward corresponding security requirements for traffic control, user-level blacklist control, SDK version management security, application-level security control, and authorization credential frequency control.

10.4.1 Flow control

To prevent the impact of abnormal burst traffic on it, the authentication platform should support traffic control plans that can intercept and limit abnormal traffic from both the client and platform.

10.4.2 User-level blacklist management and control

The authentication platform shall implement user blacklist control strategies for scenarios such as users requesting prohibition of an identity authentication service based on HTTP header enrichment technology.

10.4.3 SDK version management security

The authentication platform should implement a service-enable-function to shut down the authentication service of a defective SDK version when necessary.

10.4.4 Application level security control

The authentication platform should implement a service-enable-function to shut down the authentication service of a service provider that abnormally calls it and prohibit unauthorized businesses from continuing to call the service.

10.4.5 Authorization credential frequency control security

The authentication platform should control the frequency of abnormal calls that frequently request authorization tokens.

Bibliography

[b-GSMA PDATA.03] GSM Association – Official Document PDATA.03 (2015), *CPAS04 authenticator options*, version 1.0.

[b-IETF RFC 6749] IETF RFC 6749 (2012), *The OAuth 2.0 authorization framework*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks, open system communications and security**
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Published in Switzerland
Geneva, 2024