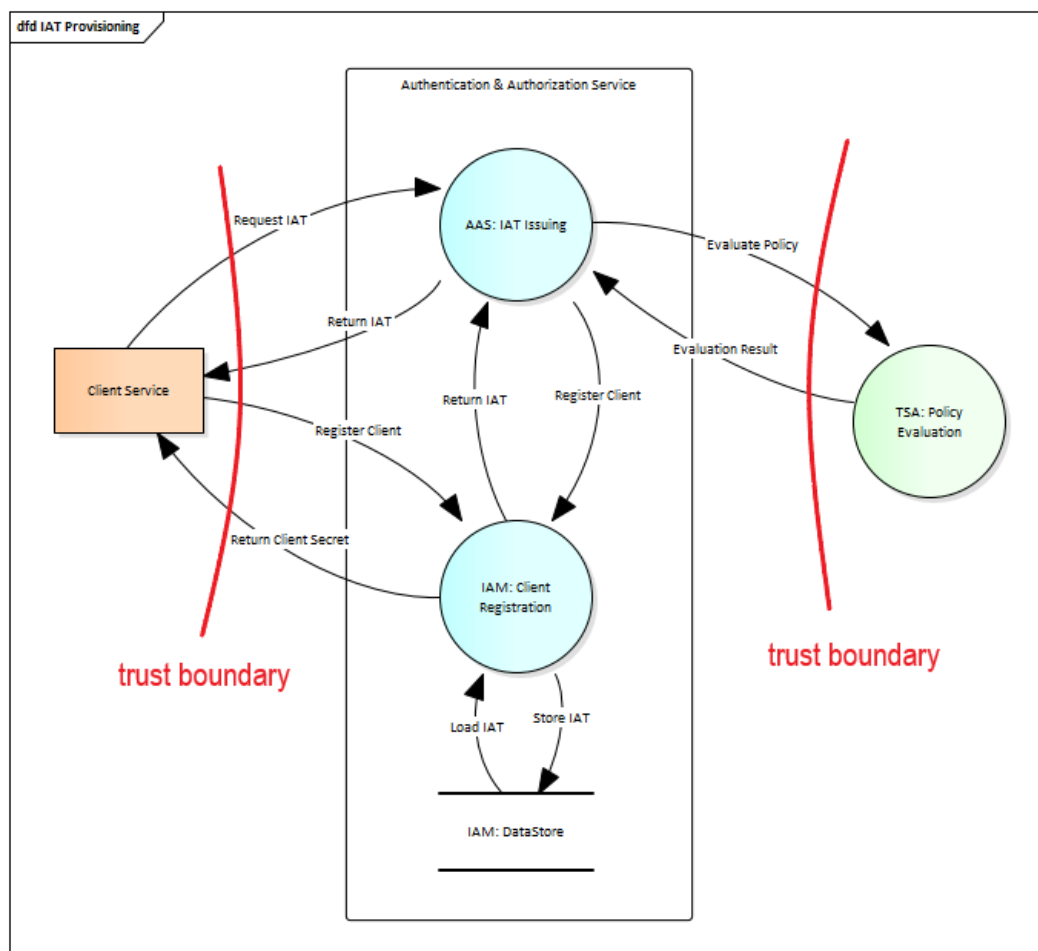


Threat Model

The following threat analysis applies to all parts of the system that are not part of the OIDC specification and therefore cannot be covered by the threat model in the "OAuth 2.0 Threat Model and Security Considerations¹" documentation.

Initial Access Token Issuing

The scenario happens between two GAIA-X domains (Client domain, SSI domain), when some application or service from Client domain wants to be registered in SSI domain and in its internal IAM Platform. To be protected with AAS and IAM, client applications must be registered in the IAM. Client can be registered in IAM using Dynamic Client Registration (DCR) protocol, but at the call to DCR endpoint Client must be authenticated. Client can be authenticated to IAM with help of Initial Access Token (IAT) provided as Bearer value in Authorization header. So, the IAT Issuing interface provides an ability for external Clients to obtain IAT to be used for authentication in subsequent DCR scenario.



Data Flows

Client Service ⇔ AAS: IAT Issuing

Information Disclosure: If an attacker would be able to access information communicated between Client and the AAS: IAT Issuing they could access the IAT token exchanged between these communication parties, thus would be able to register a malicious service in the system.

¹ <https://datatracker.ietf.org/doc/html/rfc6819>

Denial of Service: Since a client service does not need to be authenticated to access this endpoint, an attacker could flood this endpoint with requests. The AAS: IAT Issuing service requests additional endpoints every time a valid request is sent to the service and would most likely flood the IAM: Client Registration with requests as well. It is not clear how large the working leverage is between the attacker's request and the work the AAS: IAT Issuing service has to do for every request, but this could lead to an amplification of the attacker induced application-level DoS.

Tempering: If an attacker is able to manipulate the data passed between the client service and the AAS: IAT Issuing, this could lead to another form of DoS attack where a client service is unable to register its service with the system.

Client Service ↔ IAM: Client Registration

Information Disclosure: If an attacker were able to access information exchanged between the client and the IAM: Client Registration, they could access the IAT token exchanged between these communication parties. This alone is not a problem, as the IAT token would be consumed during the actual client registration process. However, if an attacker is also able to manipulate the data in the communication, he could manipulate the actual IAT token (so that it is not accepted by IAM: Client Registration) and use the unmodified IAT token to register a malicious service in the system itself.

Denial of Service: Since the IAT token is equivalent to authentication, this endpoint should not be vulnerable to an application-level DoS attack.

Tempering: By tampering with the data transmitted between Client Service and IAM: Client Registration, an attacker could manipulate the data about which service should be registered in the system and thus cause the registration of a potentially malicious service. In addition, by manipulating the data, a type of DoS attack is also possible, in which the attacker modifies the transmitted data in such a way that a registration of the client service fails. In combination with an information disclosure attack, this can also lead to the IAT token being compromised.

AAS: IAT Issuing ↔ TSA: Policy Evaluation

Information Disclosure: Even though only publicly known and accessible information is transmitted between these endpoints, it would still be a privacy issue if an attacker would be able to get access to the communicated information

Denial of Service: Since this endpoint is only accessible after mutual authentication, there is no threat of an application-level DOS attack.

Tempering: By tampering with the data transmitted between Client Service and IAM: Client Registration, an attacker could manipulate the evaluation results returned by the TSA: Policy Evaluation and thus would be able to register a potentially malicious client or deny the registration of a legitimate client service.

Entities

Client Service

Spoofing: Spoofing the client could only be achieved with knowledge of the IAT token. An attack would either have to guess the IAT token or use another vulnerability to gain knowledge of the IAT token.

Repudiation: Since the IAT token is consumed when used and the interaction is logged, repudiation in this case is not a threat.

TSA: Policy Evaluation

Spoofing: Should an attacker succeed in spoofing the "TSA: Policy Evaluation", the attacker may be able to register a potentially malicious client or deny registration of a legitimate client service by returning the appropriate evaluation results.

Repudiation: The TSA: Policy Evaluation could potentially send a wrong evaluation result for the credentials of a request and deny this action afterwards. In this case a legitimate user could be barred from interacting with the system or a malicious user could gain access to the system without the ability to pinpoint where the wrong authentication happened.

Processes

AAS: IAT Issuing

Spoofing: If an attacker could spoof the IAT Issuing process, they could intercept an IAT request and piggyback on the real authentication of a client service.

Tempering: If an attacker can alter data in this process, he could approve or deny the registration for any client service.

Repudiation: If the logs generated by the system are stored on the same system, the logs could be corrupted in the event of a system failure or maliciously manipulated by an attacker.

Information disclosure: An attacker that can access the data processed by the process, is able to steal an IAT token and use the token to register a malicious client before the legitimate user can use the token.

Denial of Service: As the process does accept unauthenticated requests that have a certain degree of leverage (a certain amount of work has to be done for every request), a possible DoS attack via request flooding could make the service unavailable. In this case no new Client Services can be registered in the system.

Elevation of Privilege: As the AAS: IAT Issuing is separated from all other processes an attacker could only access data and functionality of the AAS: IAT Issuing, even if they are able to elevate their privileges.

IAM: Client Registration

Spoofing: If an attacker is able to spoof the IAM: Client Registration process, they could gain access to all client information send by the AAS: IAT Issuing process to the IAM: Client Registration process.

Tempering: If an attacker can alter data in this process, he could approve or deny the registration for any client service.

Repudiation: If the logs generated by the system are stored on the same system, the logs could be corrupted in the event of a system failure or maliciously manipulated by an attacker.

Information disclosure: In case of an information disclosure attack being carried out, the attacker would gain access to the various client service registration information handled by this process. Additionally, they could access already created IAT tokens and thus would be able to register a potentially malicious client and deny the registration of a legitimate client service.

Denial of Service: As this process only accepts authenticated requests, there is only a low risk of an application-level DoS attack.

Elevation of Privilege: As the IAM: Client Registration is separated from all other processes an attacker could only access data and functionality of the IAM: Client Registration process, even if they are able to elevate their privileges.

Data Store

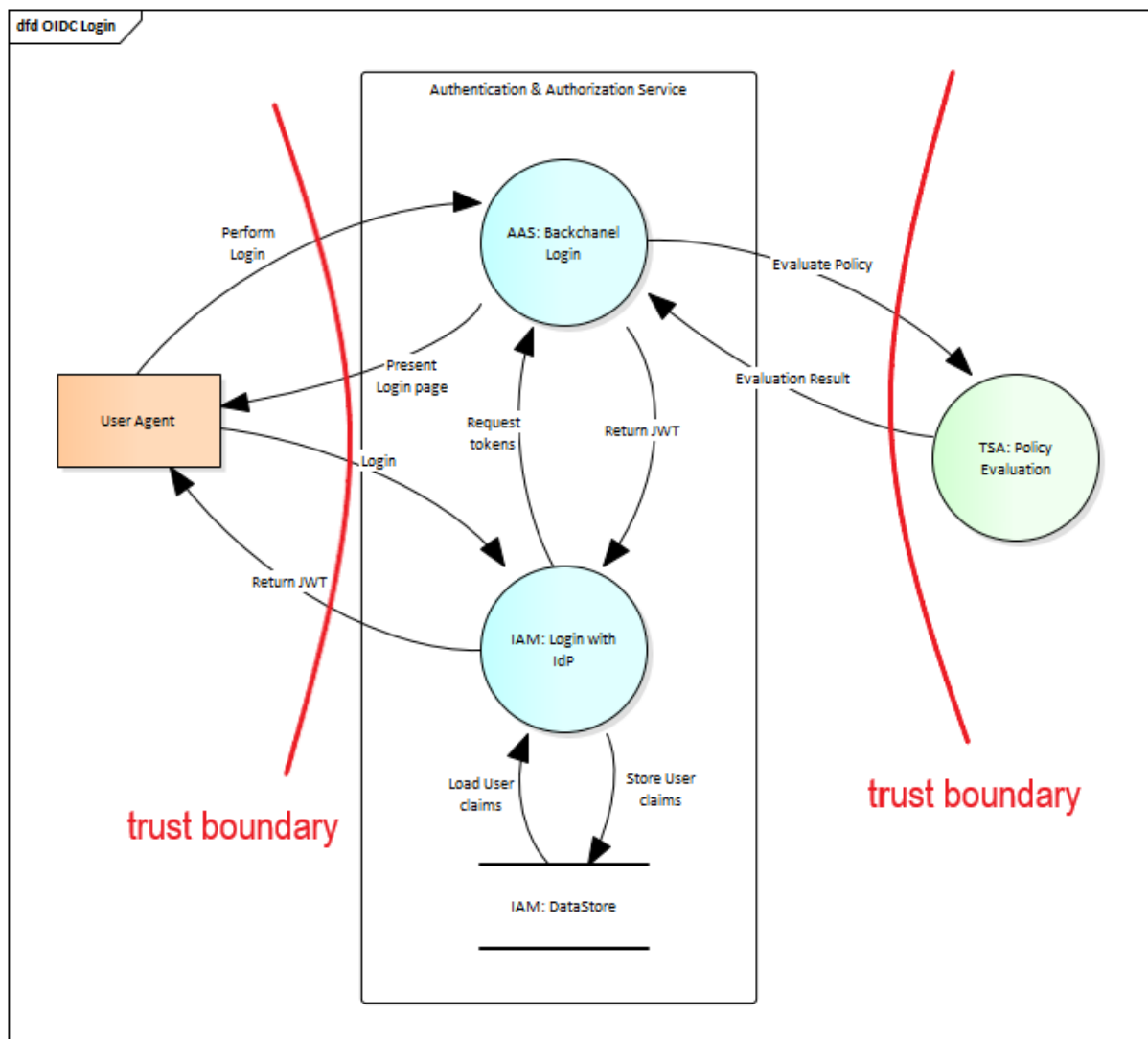
IAM: Data Store

Tempering: By tempering with the IATs stored in the IAM: Data Store an attacker could effectively make it impossible to register new client services within the system.

Information disclosure: An attacker that would be able to access data stored on the IAM: Data Store would be able to access client registration information and IATs and thus would be able to register possible malicious client services.

Denial of Service: In case of a DoS attack on the IAM: Data Store the system would not be longer able to accept client registrations, as it would not be possible to store the respective IATs.

SSI Backchannel login



Most of this use case is defined by the OIDC standard, the corresponding threat assessment, including various countermeasures, is detailed in the "OAuth 2.0 Threat Model and Security

Considerations". Therefore, this threat assessment only considers the parts that are not covered in the OIDC standard and thus still need to be addressed. Specifically the flows that need to be modeled are:

- AAS: Backchannel Login ⇔ TSA: Policy Evaluation
- User Agent ⇔ AAS: Backchannel Login (partially)

Data Flows

User Agent ⇔ AAS: Backchannel Login

Information Disclosure: If an attacker would be able to access information communicated between User Agent and the AAS: Backchannel Login he would be able to steal the RequestID. With this information he could be able to authenticate as the respective user.

Denial of Service: Since a User Agent does not need to be authenticated to access this endpoint, an attacker could flood this endpoint with requests. The AAS: Backchannel Login service requests additional endpoints every time a valid request is sent to the service and would most likely flood the TSA: Policy Evaluation with requests as well. It is not clear how large the working leverage is between the attacker's request and the work the AAS: Backchannel Login service has to do for every request, but this could lead to an amplification of the attacker induced application-level DoS.

Tempering: If an attacker is able to manipulate the data passed between the User Agent and the AAS: Backchannel Login, he could manipulate the link contained in the QR code. This could either lead to a DoS for the user or the user could be tricked to authenticate with a malicious endpoint under the control of the attacker.

AAS: Backchannel Login ⇔ TSA: Policy Evaluation

Information Disclosure: Even though only publicly known and accessible information is transmitted between these endpoints, it would still be a privacy issue if an attacker would be able to get access to the communicated information.

Denial of Service: Since the service is only accessible with mutual authentication, there is no threat of an application-level DOS attack.

Tempering: By tampering with the data transmitted between User Agent and TSA: Policy Evaluation, an attacker could manipulate the evaluation results returned by the TSA: Policy Evaluation and thus would be able to authenticate a potentially malicious User Agent or deny the authentication of a legitimate User Agent.

Entities

User Agent

Spoofing: Spoofing the User Agent could only be achieved with knowledge of the RequestID. An attack would either have to guess the RequestID or use another vulnerability to gain knowledge of the RequestID.

Repudiation: Since the RequestID is consumed when used and the interaction is logged, repudiation in this case is not a threat.

TSA: Policy Evaluation

Spoofing: Should an attacker succeed in spoofing the "TSA: Policy Evaluation", the attacker may be able to authenticate a potentially malicious user or deny authentication of a legitimate user by returning the appropriate evaluation results.

Repudiation: The TSA: Policy Evaluation could possibly send a false evaluation result for the credentials of a request and subsequently deny this action. In this case, a legitimate user could be excluded from interacting with the system, or a malicious user could gain access to the system without the ability to determine where the false authentication occurred.

Processes

AAS: Backchannel Login

Spoofing: If an attacker could spoof the AAS: Backchannel Login process, they could intercept an authentication request and piggyback on the real authentication of a User Agent to authenticate their own malicious User Agent.

Tempering: If an attacker can alter data in this process, he could approve or deny the authentication for any User Agent.

Repudiation: If the logs generated by the system are stored on the same system, the logs could be corrupted in the event of a system failure or maliciously manipulated by an attacker.

Information disclosure: An attacker that can access the data processed by this process, is able to steal the RequestID, auth_code or the authentication token used to access resources in the scope of the respective user.

Denial of Service: As the process does accept unauthenticated requests that have a certain degree of leverage (a certain amount of work has to be done for every request), a possible DoS attack via request flooding could make the service unavailable. In this case no new Client Services can be registered in the system.

Elevation of Privilege: As the AAS: Backchannel Login is separated from all other processes an attacker could only access data and functionality of the AAS: Backchannel Login, even if they are able to elevate their privileges.

IAM: Login with IDP

Spoofing: If an attacker is able to spoof the IAM: Login with IDP process, they could gain access to any auth_codes send by the User Agent and use this to authenticate a malicious User Agent with the system. (either by using the auth_code with the real IAM: Login with IDP process or try to exchange the auth_code with the AAS: Backchannel Login process for a token themselves)

Tempering: If an attacker can alter data in this process, he could approve or deny the authentication for any User Agent.

Repudiation: If the logs generated by the system are stored on the same system, the logs could be corrupted in the event of a system failure or maliciously manipulated by an attacker.

Information disclosure: An attacker that can access the data processed by this process, is able to steal the authentication token used to access resources in the scope of the respective user.

Denial of Service: As this process only accepts authenticated requests, there is only a low risk of an application-level DoS attack.

Elevation of Privilege: As the IAM: Login with IDP is separated from all other processes an attacker could only access data and functionality of the IAM: Login with IDP process, even if they are able to elevate their privileges.

Data Store

IAM: Data Store

Tempering: By tempering with the User Claims stored in the IAM: Data Store an attacker could manipulate the stored PII or other attributes and possibly disrupt or misuse the service.

Information disclosure: An attacker that would be able to access data stored on the IAM: Data Store would be able to access PII of all users registered with the system.

Denial of Service: In case of a DoS attack on the IAM: Data Store the system would not be longer able to process authentication actions, thus disrupting the whole SSI backchannel login.