R-A&B: The technical contribution of our work.

There have already been such works focusing on the privacy issues in the SSO systems, such as SPRESSO and PPID. However, all the existing work only focus on one side privacy, IdP-based tracing or RP-based linkage. Moreover, the current existing solutions are mutually exclusive, which means they cannot be simply integrated together. We are the first to comprehensively analyze the privacy problem in SSO and propose the trapdoor-based transformation.

R-B&D: The importance of protecting users’ privacy.

The privacy issues preventing the IdP-based tracing are well motivated, as there have already been the works focusing on how to prevent IdP from knowing the RP’s identity, such as SPRESSO[1] and BrowserID[2].

However, RP-based identity linkage is another privacy problem is SSO systems. In some popular SSO specifications (e.g., OIDC and SAML), the PPID (i.e., IdP provides different user IDs for different RPs) has been explicitly claimed as the privacy consideration to protect the user from a possible correlation among RPs. Many identity management systems by large companies, such as Active Directory Federation Services (AD FS) and Oracle Access Management, have already supported PPID [3][4]. Moreover, some identity service working on identity services, such as NORDIC APIS and CURITY, also suggest that the PPID should be used in SSO systems to protect user’s privacy [5][6]. Therefore, we consider the prevention from RP-based identity linkage should be an important point in SSO systems.

R-A: The impersonation of adversary caused by the requirement of uniqueness of IDs.

An adversary (acts as an RP) may negotiate an PID\_RP with an honest user, and this adversary (acts as a user) negotiates the same PID\_RP with an honest RP. Only in this situation, the adversary can use the identity proof received from honest user to impersonate this user at the honest RP. As the PID\_RP is finally decided by the user generated N\_U, the possible attack is that, the adversary receives an identity proof with PID\_RP, and he may try to negotiate the same PID\_RP with another honest RP. However, adversarial user only provides the N\_U, and computing N\_U from ID\_RP and PID\_RP is infeasible, so that he can only guess the correct N\_U in brute force, which is infeasible.

R-B: How does the IdP select which RPs it can provide users information to.

It current popular SSO system, the user information exposed to RP from IdP should under the explicit user consent. For example, the authentication request defined by OpenID Connect protocol should carry the parameter scope, which represents the required user information (e.g., scope=phonenumber email). IdP would display the required information on its webpage and user need click the confirm button. The similar function is optional in UPPRESSO as it is compatible with OpenID Connect.

However, in this paper we only consider the privacy-sensitive user, who do not want to expose any sensitive unique information (such as email) identifying a user. In this situation, other insensitive information can be provided to any qualified RPs, where the validation of RP is confirmed by Cert.

R-B: How to prevent malicious RP from tricking users by letting them download a script that generates specific N\_U.

The malicious RP cannot let user download the malicious script from IdP as IdP is considered always honest. The malicious RP can lead user to download the script form a malicious server, however, due to the same origin policy the script cannot visit the IdP server (for example, obtain the identity proof from IdP server). Therefore, the login process cannot go on with the malicious script.

R-B&D: The overhead of UPPRESSO.

According to [], we can find that the users’ tolerance on internet delay is . And the affordable internet round trip time is even 200ms. Therefore, we consider the latency of UPPRESSO is affordable. The extra time cost of UPPRESSO is the trade for user privacy. Moreover, most of the extra time cost takes place at the user side, for example, downloading scripts, opening new window and transmitting message between scripts. UPPRESSO only requires few modpow computes on IdP and RP server, where the single compute takes no more than 10ms, so that there is not significant time cost at either RP or IdP side.

And we are working on the experiment evaluating throughput rate of RP and IdP server. The result is to be added in the next version of paper.

R-C: Using elliptic curves.

The solution based on elliptic curves may be pretty good and can achieve the same function as current design. However, we only adopt the solution based on discrete logarithm problem to get an early verification version quickly. Our main contribution in this paper is proposing the three transformation functions based on the trapdoor existing algorithm, which can be implemented based on either discrete logarithm problem or elliptic curves. And it can also satisfy the same security requirement as elliptic curves. Moreover, the time cost of modpow computes are only accounted for a small part of overhead, so that it would not bring the significant performance improvement by using elliptic curves. We are working on the expanding the protocol with elliptic curves which need only the minor modification in the implementation. And we will show it on the next version of paper.

R-C: Adversary cannot obtain information of x from g^x in UPPRESSO.

UPPRESSO can achieve the same level of security as DDH assumption because the exponent and base values can be never controlled by an adversary. The important values in the system are ID\_RP, ID\_U and N\_U. That is, ID\_RP is generated by the honest IdP and protected by Cert, so that it cannot be controlled by RP. ID\_U is provided by IdP and N\_U is provided by user. It would not bring any benefits to IdP and user by providing insecure ID\_U or N\_U.

Therefore, this attack proposed by reviewer is not possible in UPPRESSO system. The ID\_RP of each RP is contained in the Cert and verified by user during the login process, so that RP cannot temper the value to make ID\_RP and ID\_RP\* in a correlated way.

R-C: Security Analysis and Threat Model

As the transformation functions in UPPRESSO are based on the discrete logarithm problem, the challenges for malicious RP and curious IdP are computing ID\_U from ID\_RP^(N\_U\*ID\_U) or ID\_RP from ID\_RP^N\_U, which are considered infeasible. Therefore, our analysis only focuses on the security of web model based on Dolev-Yao style model, which has been widely used in SSO system analysis [][][].