JASF: Jasta Security Framework *

Surendranath Chowdary Chandra
Jasta Techologies LLP
Bangalore, India - 560 078
surendranathchowdaryc@gmail.com

Ravindranath Chowdary C
Jasta Techologies LLP
Bangalore, India - 560 078
ravindranathchowdaryc@gmail.com

ABSTRACT

JASM is a model designed to increase the security level in authentication systems. It uses IP Address of the user in the authentication process to enhance the security.

Keywords

Security, Authentication, Phishing

1. INTRODUCTION

On-line security plays a vital role to prevent users sensitive data from falling into wrong hands. JAsta Security Model(*JASM*) can be used for providing higher degree of security to system/application where users authentication is required. By using this model, the problem of phishing[1] can be solved to a greater extent. This model can be deployed in banking systems, mailing systems, and in high security zones.

2. EXISTING SECURITY MODELS

Here we mention two most commonly used security models.

2.1 Existing Security Model 1(*ESM1*)

This security model typically requires a unique username and a password to login to an application. The attacker could easily get the user's credentials through phishing techniques and the user could end up being totally unaware of the event at all. Example for such systems could be email systems, where user keys in his userid and password to login to his account.

2.2 Existing Security Model 2(*ESM2*)

In systems implementing *ESM2*, initially user will be prompted for his username and password and these details will be sent to the application. If they are verified to be correct, then as part of the second step, the user will be prompted for verifying the One-time Verification Passcode(*OTP*) sent to the user's registered mobile number with the application. On successful verification, the user will be able to login to his account. Example for such system is a 2-step verification mechanism provided by Gmail$^{\text{TM}}$for (Google Inc.). *ESM2* is more secure than *ESM1*.

2.2.1 Breaking of ESM2

ESM2 is more secure than *ESM1*, because it uses 2-Step 2-Channel verification. Though *ESM2* is secure to some extent, it is having disadvantages. Through man-in-the-middle attack(*MITM*), the attacker can easily get hold of user's password as well as his *OTP*. In this way *ESM2* can be completely compromised.

3. JASTA SECURITY MODEL(*JASM*)

ESM2 depends on what user knows(password) and what user receives(*OTP*) to what he has(registered mobile) to provide security. *JASM* is also a 2-Step 2-Channel verification model but it attempts to increase the level of security of the authorisation systems.

3.1 Terminology

- *Secret Code(SC)*: *SC* can be user's lucky number or a favorite dish or a common phrase or anything that user intends to maintain as a secret.

- *OTPSC*: *OTP* followed by *SC* is called *OTPSC*. For example, if *OTP* sent by the system is "12345" and *SC* of the user is "*ABC*", then *OTPSC* is "12345*ABC*".

- *Account Access IP(AAIP)*: Client-side IP Address from which the request for accessing user's account was made.

3.2 Assumptions

- There will be seamless internet connectivity

- There will be seamless mobile connectivity

- With each user account two mobile numbers are registered. Mobile number to which all account access related activities are to be sent by default is registered as *primary mobile number*. Another mobile number is registered as *secondary mobile number*, so that it can act as primary mobile number in case of loss/non-accessibility of the *primary mobile number*.

[1] http://en.wikipedia.org/wiki/Phishing

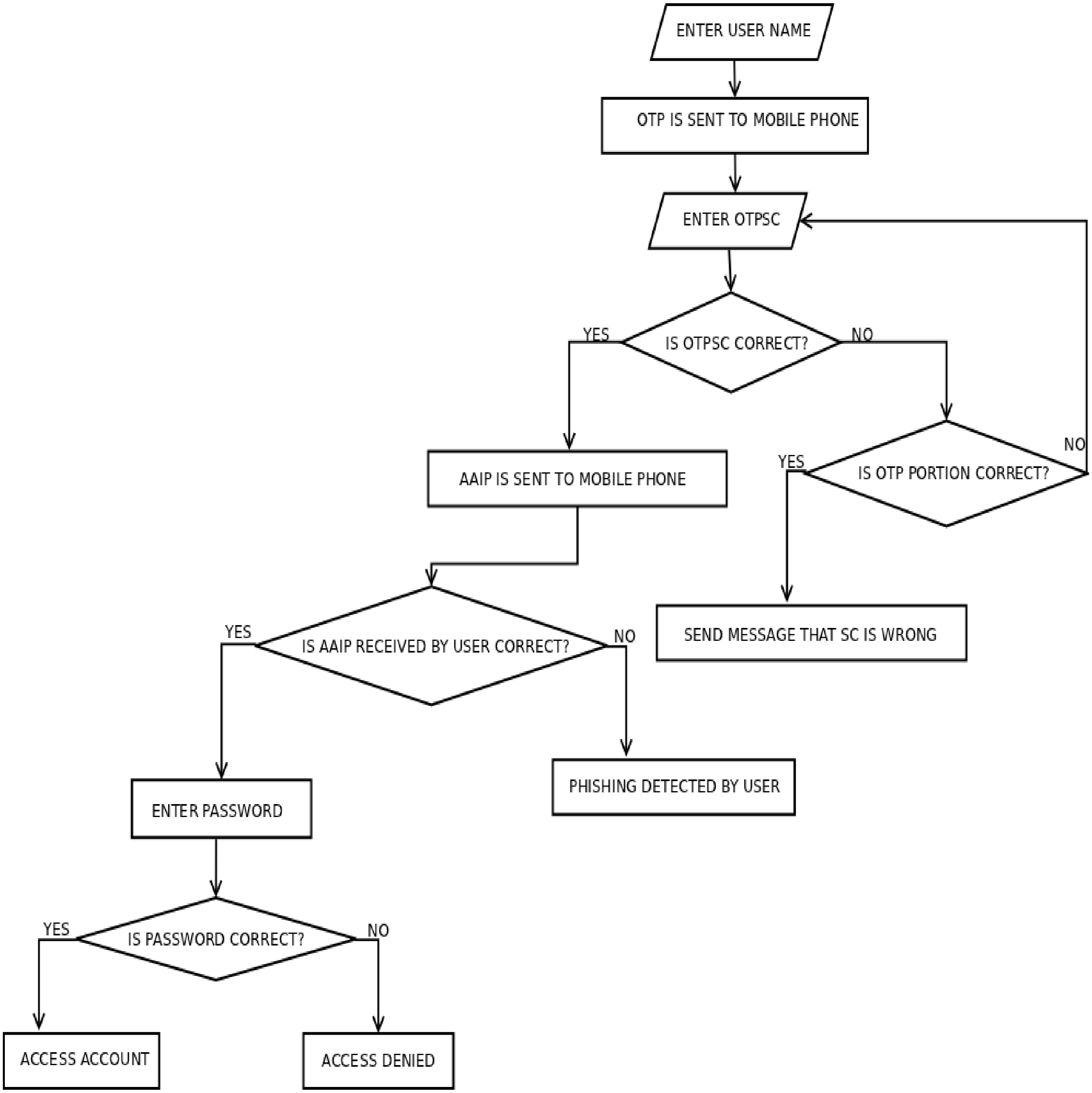

Figure 1: Flowchart of *JASM*

3.3 Description of *JASM*

The flowchart of *JASM* is represented in *Figure 1.*

1. User enters username to login to his account.

2. The system, sends *OTP* to the user's *primary mobile number* to access his account. If the user selects *secondary mobile number* to receive *OTP*, then the *OTP* that was already sent to the *primary mobile number* will be re-sent to the *secondary mobile number.*

3. User will enter *OTPSC* of the corresponding account.

4. System verifies the *OTPSC.*
   Here there will be 3 possible cases.

   - *Case 1*: if the *OTP* portion is wrong, then the system will prompt for re-entering *OTPSC.*
   - *Case 2*: if the *OTP* portion is correct but *SC* portion is wrong, then there may be a possibility that attacker got hold of user's *OTP* and he is trying to crack the *SC.* System will test if such scenario is repeated for a certain threshold. If that threshold is exceeded, then it will send *AAIP* stating that *SC* is wrong to both the user's registered mobile numbers. Thus, it alerts the user.
   - *Case 3*: if *OTPSC* is entered correctly, then *AAIP* is sent to either user's *primary* or *secondary* mobile number based on what he has chosen. Then user verifies the correctness of *AAIP* and can access his account by entering the password.

3.3.1 Handling of OTP

- User will get *OTP* once he enters his username. *OTP* will be valid only for a certain time-frame(*e.g., 2hrs, 6hrs*). If *OTP* is not used in that time-frame, user will get a new *OTP* when he tries to access his account next time.

- If someone types in other user's username, the mobile will not be flooded with *OTPs.* User will get a new *OTP* only after using the current *OTP* or after the *OTP's* time-frame expires, whichever is earlier.

- In case of loss/non-accessibility of both the user's registered mobiles, then spl-passcodes should be available with the user as in 2-step verification mechanism provided by Gmail$^{\text{TM}}$for (Google Inc.). So the user enters spl-passcode along with his *SC*. Even in this case, *AAIP* will be sent to the user's registered mobile as a security measure.

3.3.2 More of AAIP

- *In the absence of client-side proxy*: The IP Address verification through *AAIP* will help the user to verify whether the *OTPSC* received by the server is from his IP.
  For *e.g.*, if the user's IP is *x.y.z* and he receives *AAIP* as *a.b.c*, then he can be sure that he is under *MITM.* On the other hand if the *AAIP* is *x.y.z*, then he can be sure that he is not under *MITM.*

- *In the presence of client-side proxy*: The IP Address verification through *AAIP* will help the user to verify whether the *OTPSC* received by the server is via his proxy itself.
  For *e.g.*, if the user's proxy IP is *x.y.z* and he receives *AAIP* as *a.b.c*, then he can be sure that he is under *MITM.* On the other hand if the *AAIP* is *x.y.z*, then he can be sure that either himself or someone behind the same proxy have given the *OTPSC* to the server. Note that in the case of a proxy, *JASM* may fail if the attacker is behind the same proxy.

4. ADVANTAGES OF *JASM* OVER *ESM2*

- All advantages of *ESM2* are applicable to *JASM.*

- As mentioned in *Section 2.2.1*, the attacker cannot get the user's password in *JASM.* The reason is, the system will verify the user through *OTPSC* entered by him and the user will make sure that the *OTPSC* received by the server is from his IP. This is verified through *AAIP* he receives. So, this mechanism acts as a *2*-way handshake, before the user enters his password. So, it results in high security.

- If the attacker gets hold of user's *OTP* then also he cannot crack *OTPSC.*

We can thus conclude that *JASM* is far more secure than *ESM2.*