Using M-Pin with Geo-location/Geo-fencing

Michael Scott

Chief Cryptographer
MIRACL Labs
mike.scott@miracl.com

1 Introduction

M-Pin is a Multi-Factor identity based authentication protocol. Secret authentication keys are issued to clients from a Distributed Trusted Authority (DTA). Typically there are two DTAs, one belonging to the customer and the other to MIRACL. Once the client has its secret, formed by adding the two components received from each DTA, it can use it to authenticate to an M-Pin server that has been issued by the DTAs with a single secret of its own. The client secret can be chopped up into individual factors, which are re-united only at the point where authentication is required.

2 Geo-Fencing

Geo-fencing is advantageous to both the server and the client. It limits the locations from which a client can successfully authenticate to a server. The server can be sure that even a valid client who is outside of the permitted area cannot successfully log in, and the client can be sure that if their device is stolen it cannot be used away from their place of work. It adds an extra factor to the authentication process.

However we should be aware that location can always be faked by the well equipped attacker. System calls to the GPS hardware can be intercepted, and false coordinates passed to the application. Therefore as an extra factor, location has limited value.

When the client registers they choose (or have chosen for them by the customer DTA) a location that will be at the centre of a square area outside of which they will not be able to authenticate. The size of this area (the length of the side of the square) is also chosen. An offset to be added to the GPS coordinates in order to centre this location in a square, and also so that the longitude and latitude of the bottom left hand corner (BLHC) can be represented as whole numbers, is stored on the device, as is a scaling value depending on the length of the side of the square.

Loosely speaking the (x, y) coordinates of the BLHC of the square can be subtracted from the client secret by the Customer DTA, before the secret is issued. Alternatively this can be done by the client themselves. Then when the client authenticates the GPS on the device is accessed, the BLHC of the square the device is in is calculated, and added back to recreate the original secret.

Recall that the client secret is of the form sA, where A is the identity of the client hashed to a point on an elliptic curve, and s is the overall TA master secret. We suggest that the location is embedded in this secret by replacing the stored client secret with $sA - (2^{32}x + 2^{64}y)A$. These offsets are chosen to ensure that there is no interaction with a PIN number which may also be subtracted from the client secret.

When the time comes to authenticate, the GPS hardware is accessed, and the (X, Y) coordinates retrieved. These are used to add back to the client secret $(2^{32}X + 2^{64}Y)A$ so it becomes $sA - (2^{32}x + 2^{64}y)A + (2^{32}X + 2^{64}Y)A$, which obviously becomes sA only if $x = X$ and $y = Y$.

3 Worked Example

Assume that the client registers their device at latitude 13.06278, longitude 80.22946. Now a 0.01 difference in either longitude or latitude is roughly 1000 meters. Assume that the client is expected to be within a 1 kilometer square, which is centred on this initial point of registration.

We calculate the offset as (.00222,-.00446), which when added to the latitude/longitude centers the client in its square. This offset is stored and the (x, y) coordinates are converted to integers by simple truncation as (1306,8022) and these values are used to modify the client secret as described above.

Now when for example the client authenticates from the location latitude 13.06567, longitude 80.23127, the offset is added to this to get latitude 13.06789 and longitude 80.22681. This is truncated to the integer coordinates (1306,8022), and since these are the same as before, they will cancel out the modification to the secret, to restore the original secret sA, which can then be used to successfully authenticate.

4 Security

The stored offset does not leak anything useful, as it represents a small shift relative to the registration location, which will not be known to an attacker who has stolen the device. In the M-Pin tradition nothing related to client location is stored on the server.

5 Deployment

The registration location can either be enforced by the Customer via the DTA under its control. Or it can be found from the client's device location at the moment of registration, by accessing its GPS hardware. The latter is certainly simpler. There are lots of code examples out there on accessing the GPS hardware from IOS and Android.