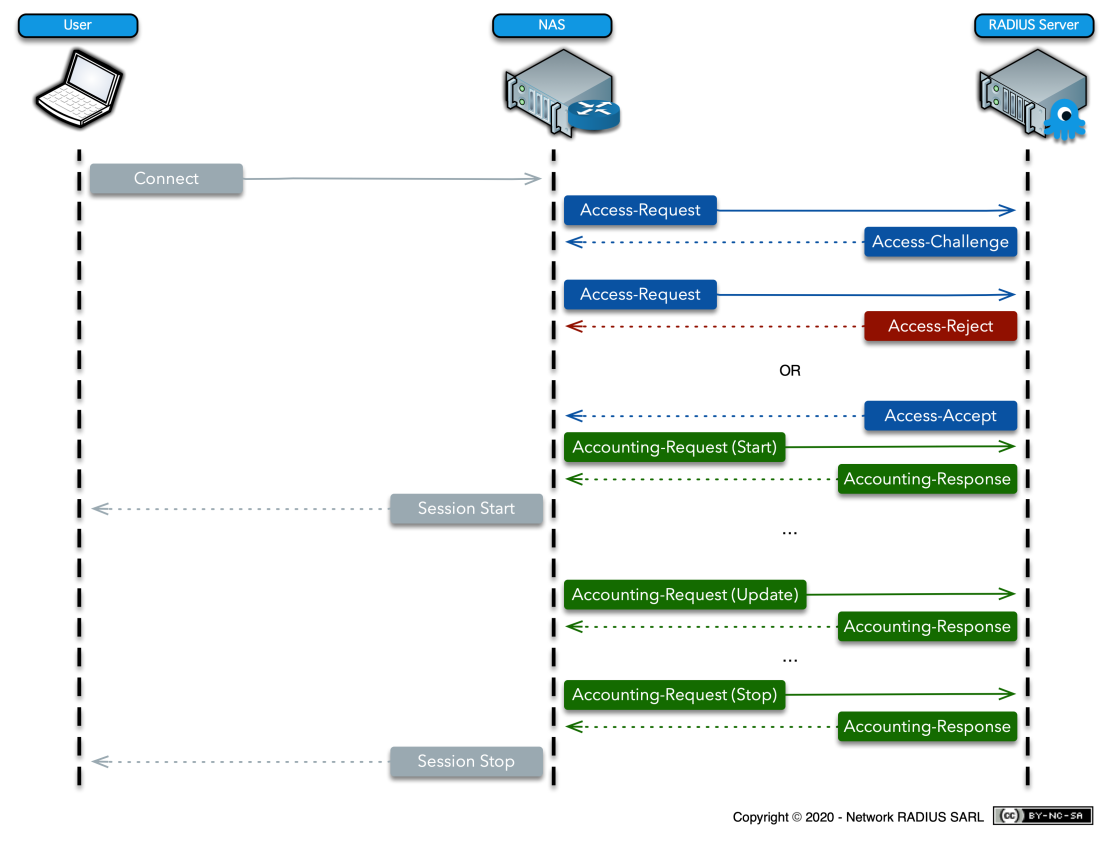
RADIUS OTP

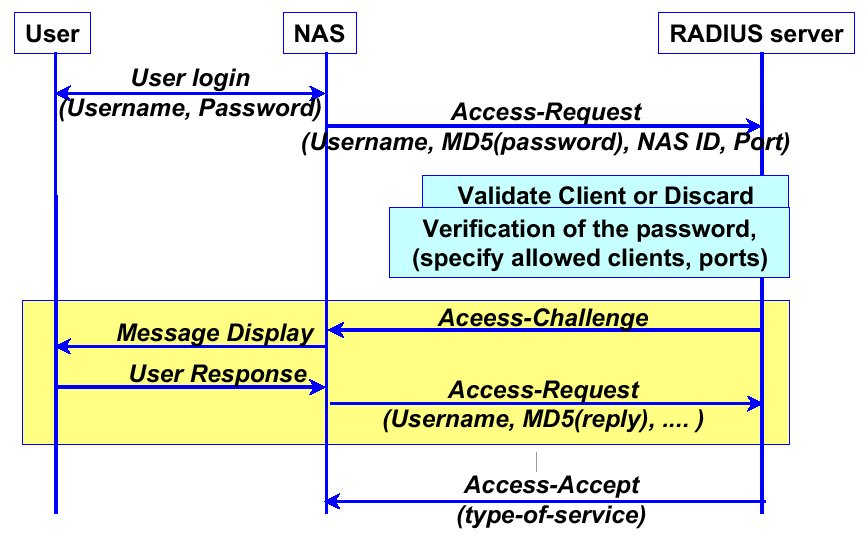
1. OTP là gì?

* Là phương thức xác thực đăng nhập một khẩu một lần và chỉ có giá trị cho một lần đăng nhập (xác thực hai nhân tố). Mục đích để đảm bảo an toàn tài khoản cho người dùng, tránh bị đánh cắp dữ liệu. Mã OTP được sinh ra ngẫu nhiên sau thời gian nhất định.

1. Radius

* Là protocol, chuyển các yêu cầu xác thực đến một hệ thống quản lí danh tính. Theo mô hình client – server. Dùng để truy cập từ xa.
* Kiến trúc: AAA: Authencation, Authorition, Accouting
* Dùng port: 1812, Accouting dùng 1813.
* Giao thức UDP, chuẩn RFC 2865
* Giao thức hỗ trợ PAP, CHAP, MSCHAP, …





1. RADIUS OTP

* Xác thực cấp hai qua giao thức Radius.
* Hoạt động:

The device Radius Client Radius Server AD, LDAP

(NAS)

Enter user + pass

Response

Response

Send Otp

First authen

waiting

Enter Otp

Response

Response

Response

Check

Send user + pass

Gg authen

SMS

Access Reject/ Access Accept

Access Reject/ Access Accept

Access-Challenge

+ Từ chối truy cập - **Access Reject**: Người dùng bị từ chối truy cập vào tất cả các tài nguyên mạng.

+ Yêu cầu gửi thêm thông tin truy cập - **Access Challenge**: Yêu cầu thông tin bổ sung từ người dùng mã OTP.

+ Chấp nhận truy cập **- Access Accept**: Người dùng được cấp quyền truy cập.

* **Hoạt động:**
* Các loại xác thực:

+ Nhập user, pass -> sau đó mới nhập otp

+ Nhập user, pass, nhập luôn otp

+ Nhập user, pass nối chuỗi với otp.

Nhập user, pass -> sau đó mới nhập otp

+ User thực hiện đăng nhập với user, pass.

+ User, pass sẽ được gửi qua Radius Client, ở đây là NAS (Netword Access Server) máy chủ truy cập mạng. NAS được tích hợp Radius.

+ NAS gửi user, pass cho Radius Server.

+ Radius Server check user, pass qua AD (LDAP, CD).

+ Sau đó gửi lại thông báo cho NAS.

+ Nếu xác thực thành công thì Radius Server gửi Access Challenge cho NAS.

+ Radius Client sẽ gửi lại cho User để yêu cầu nhập mã OTP.

+ User truy cập bên thứ 3 lấy mã otp và nhập mã, gửi qua NAS

+ NAS gửi qua bên Radius server để check và gửi lại phản hồi.

+ Nếu thành công người dùng được phép truy cập.

Nhập user, pass, nhập luôn otp

+ User truy cập bên thứ 3 lấy mã otp

+ User thực hiện đăng nhập với user, pass, nhập mã OTP

+ User, pass sẽ được gửi qua Radius Client, ở đây là NAS (Netword Access Server) máy chủ truy cập mạng. NAS được tích hợp Radius.

+ NAS gửi user, pass cho Radius Server.

+ Radius Server check user, pass qua AD (LDAP, CD).

+ Nếu xác thực thành công thì Radius Server sẽ check mã Otp.

+ Sau đó gửi lại thông báo cho NAS. NAS gửi lại cho User.

+ Nếu thành công người dùng được phép truy cập.

Nhập user, pass nối chuỗi với otp.

+ User truy cập bên thứ 3 lấy mã otp

+ User thực hiện đăng nhập với user, pass, nhập mã OTP

+ User, pass sẽ được gửi qua Radius Client, ở đây là NAS (Netword Access Server) máy chủ truy cập mạng. NAS được tích hợp Radius.

+ NAS gửi user, pass cho Radius Server.

+ Radius Server tách chuỗi pass để lấy pass và otp, sau đó check user, pass qua AD (LDAP, CD).

+ Nếu xác thực thành công thì Radius Server sẽ check mã Otp.

+ Sau đó gửi lại thông báo cho NAS. NAS gửi lại cho User.

+ Nếu thành công người dùng được phép truy cập.

* Radius Server: Server Radius Free

+ Host

+ Port: default: 1812

+ Secret: key

+ Đăng ký Radius Otp với gg authen:

Mã QR

Xác thực với Google Authencation

* Với google authencation, cơ chế sinh mã sẽ cần một khóa bí mật, thời gian vào hiện tại và hàm mã hóa.
* Hàm mã hóa sử dụng HMAC-SHA1: thuật toán:  hmac = SHA1(secret + SHA1(secret + input))
* original\_secret = xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx

secret = BASE32\_DECODE(TO\_UPPERCASE(REMOVE\_SPACES(original\_secret)))

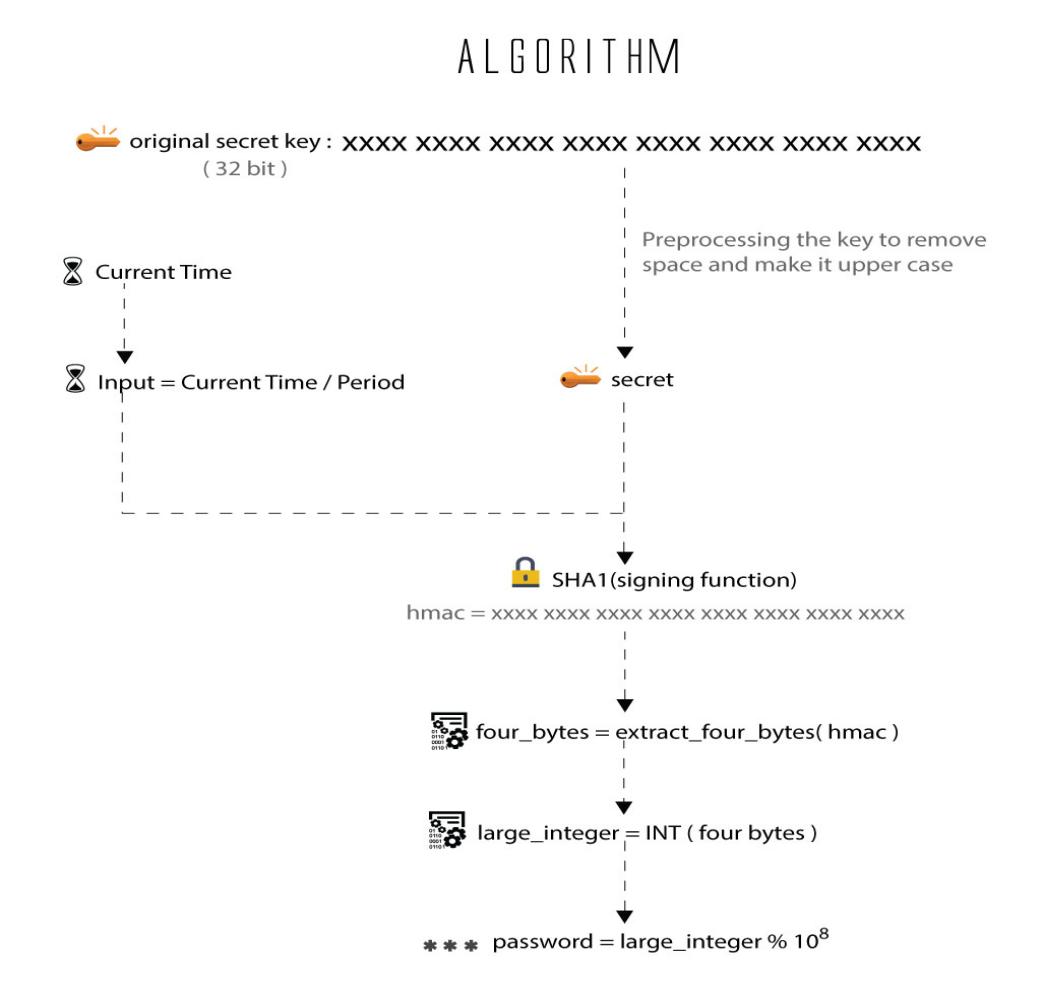
input = CURRENT\_UNIX\_TIME() / 30

hmac = SHA1(secret + SHA1(secret + input))

four\_bytes = hmac[LAST\_BYTE(hmac):LAST\_BYTE(hmac) + 4]

large\_integer = INT(four\_bytes)

small\_integer = large\_integer % 1,000,000



* *Mã QR sẽ được gen từ original\_secret, hoặc hiển thị bản rõ để có thể quét/nhập trên thiết bị.*
* Key original\_secret được lưu ở radius server khi người dùng click vào bật gg authencation.
* Radius server sẽ xử lí hàm mã hóa để lấy được giá trị small\_integer và so sánh với input người dùng nhập vào.

+ Đăng ký với SMS:

Xác thực với SMS

SĐT

* Khi người dùng bật xác thực Radius Otp với SMS thì số điện thoại người dùng sẽ lưu tại Radius Server.
* Sau khi login, Radius server sẽ random tạo ra mã Otp có thể lưu mã Otp này tại Server Radius và có thời gian hết hạn là 30s-1p. Radius server xử lí hàm gửi mã Otp về SMS theo số điện thoại được lưu tại Radius Server của người dùng. Người dùng nhập mã, Radius Server so sánh với giá trị đã lưu tại Cookie hoặc LocalStorage rồi phản hồi cho Radius Client đưa lại cho người dùng.

**+ Đăng ký với LoginTC:**

* Đăng ký tài khoản LoginTC admin.
* Lấy KEY API
* Tạo domain, tạo Application.
* Lấy Domain ID.

1. Dùng Radius OTP với LoginTC để xác thực 2 bước với NAS (bên thứ 3)

Thực hiện cầu hình:

* Setup: <https://www.watchguard.com/help/docs/help-center/en-US/Content/Integration-Guides/General/LoginTC.html>
* <https://www.logintc.com/docs/connectors/radius.html#configuration>
* <https://www.logintc.com/resources/docs/radius.html>
* LoginTC Radius Connectors kết nối với AD, LDAP.
* Tạo user (tên user cùng tên user trong AD).
* Tạo Issue Token, sinh ra 1 mã 10 kí tự, người dùng sẽ nhập mã này vào trong App LoginTC trên thiết bị. Lúc này server sẽ lưu lại tên thiết bị nhập mã 10 kí tự,
* Test trên hệ thống, LoginTC sẽ kiểm tra user, pass trong AD và chờ đợi phản hồi người dùng chấp nhận thông báo đẩy từ thiết bị, hoặc dùng mã Otp được sinh ra.
* Cấu hình AP hoặc NAS theo LoginTC Radius Connectors với ip server radius, port, pass secret.

1. Dùng LoginTC với Rest Api để xác thực dùng như Google Authencation.

* Dùng Rest Api của LoginTC để gửi request: <https://cloud.logintc.com/api/>

Hoặc mã 10 chữ số

Mã QR

Xác thực với LoginTC

* Khi người dùng đăng ký sử dụng LoginTC làm yếu tố xác thực cấp 2 thì hệ thống sẽ gửi 1 request api để thêm User vào Domain của LoginTC. Với 1 hệ thống thì đăng ký và sử dụng 1 domain trên LoginTC.

POST: /users

curl -X POST https://cloud.logintc.com/api/users \

-d '{"username" : "john.doe", "name" : "John Doe", "email" : "john.doe@example.com"}' \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"'

* Với thông tin của User gồm Username, name, email.
* Sau khi post xong thì sẽ nhận được 1 response chứa ID của User.
* Sau đó, chúng ta sẽ tạo Tokens của User để nhập vào thiết bị.

PUT /domains/{domainId}/users/{userId}/token

curl -X PUT https://cloud.logintc.com/api/domains/9120580e94f134cb7c9f27cd1e43dbc82980e152/users/8c184f495a5b7b6e9ed732f2ce3c67e310806f38/token \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"' \

-H 'Content-Length: 0'

* Chúng ta sẽ nhận được 1 response chứa mã code 10 số.
* Xử lí mã này thành QR hoặc User sẽ nhập trực tiếp trên thiết bị tại LoginTC.
* Tiếp tục tạo mã cá nhân cho người dùng nếu không thể truy cập thiết bị, mã truy cập này 9 số thay cho mã OTP (setup số lần và thời hạn của mã).

POST/api/users/{userId}/bypasscodes

curl -X POST https://cloud.logintc.com/api/users/8c184f495a5b7b6e9ed732f2ce3c67e310806f38/bypasscodes \

-d '{"usesAllowed" : "5", "expirationTime" : "43200"}' \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"'

* Lưu DomainID, KEY API tại Radius Server.
* Khi người dùng Login vào hệ thống (gửi User, Pass) lên NAS. NAS chuyển tiếp lên Radius Server trong Access Request bao gồm một số trường thông tin như Username, User-password, NAS ID, NAS port. Radius Server kiểm tra trong cơ sở dữ liệu như AD, LDAP hay chính Radius Server.
* Nếu sai thì gửi về cho NAS 1 Access Reject từ chối truy cập, NAS gửi thông báo cho User.
* Nếu thành công, Radius Server gửi lại cho NAS 1 yêu cầu xác minh thêm Access Challenge. Yêu cầu Access Challenge có thể chứa một thông điệp văn bản sẽ hiển thị bên máy người dùng. Nếu máy User nhận được yêu cầu và hỗ trợ cơ chế thách thức / phản hồi ( challenge/response) thì đoạn thông điệp sẽ hiển thị và yêu cầu User phản hồi lại hệ thống.
* User sẽ chọn 1 trong các loại xác thực để thực hiện việc xác minh.
* Dùng mã OTP.
* Dùng mã thông báo đẩy.
* Trong trường hợp người dùng không thể truy cập thiết bị thì sẽ gửi về email của người dùng. (nhập email hoặc gửi yêu cầu, Radius Server sẽ xử lí gửi mã về email với api có sẵn).
* Dũng mà cá nhân.
* Sau khi nhận được các phản hồi từ phía User, Radius client tiến hành gửi lại Access Request ban đầu với một số ID khác và trường user-password được thay thế bằng thông tin phản hồi của người dùng.
* Radius Server xử lí mã OTP nhận được:
* Dùng các thông tin của User để xử lí:
* Dùng mã thông báo đẩy:

POST/domains/{domainId}/sessions

curl -X POST https://cloud.logintc.com/api/domains/9120580e94f134cb7c9f27cd1e43dbc82980e152/sessions \

-d '{"user" : {"username" : "john.doe"}, "attributes" : []}' \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"'

hoặc

curl -X POST https://cloud.logintc.com/api/domains/9120580e94f134cb7c9f27cd1e43dbc82980e152/sessions \

-d '{"user" : {"id" : "8c184f495a5b7b6e9ed732f2ce3c67e310806f38"}, "attributes" : []}' \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"'

* Phản hồi nhận được 1 Session, Session này sẽ được xử lí tại Radius Server, gửi Request liên tục để đợi phản hồi trong thời gian nhất định.

GET/domains/{domainId}/sessions/{sessionId}

curl -X GET https://cloud.logintc.com/api/domains/9120580e94f134cb7c9f27cd1e43dbc82980e152/sessions/fcbdc4c271c889825d8338d2d8f10b6e5e95c171 \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"'

* Nếu được chấp nhận kết nối, Radius Server gửi 1 Access Accept cho NAS, NAS gửi cho User.
* Dùng gửi Email:

POST/domains/{domainId}/users/{userId}/otp

curl -X POST https://cloud.logintc.com/api/domains/9120580e94f134cb7c9f27cd1e43dbc82980e152/users/8c184f495a5b7b6e9ed732f2ce3c67e310806f38/otp \

-d '{"deliverableOTPType":"EMAIL"}' \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"'

* Dùng mã OTP:

POST /domains/{domainId}/sessions

curl -X POST https://cloud.logintc.com/api/domains/9120580e94f134cb7c9f27cd1e43dbc82980e152/sessions \

-d '{"user" : {"id" : "8c184f495a5b7b6e9ed732f2ce3c67e310806f38"}, "otp" : "123456"]}' \

-H 'Authorization: LoginTC key="YOUR\_API\_KEY"'

*~~+ Đăng ký với RSA SecurID:~~*

* *~~Là thẻ cứng chứa mã otp, mua thẻ do nhà cung cấp tạo. Quy trình tạo mã dựa trên TOTP với thời gian là 60s do nhà cung cấp phát triển.~~*
* *~~Sau khi có thẻ cứng nhập mã seri vào trang web để cập nhật thông tin và mã PIN.~~*
* *~~Sau khi cập nhật xong thì sẽ được gửi về email đã đăng kí 1 đường dẫn URL, thiết bị sẽ nhập đường dẫn hoặc mã QR để update trên thiết bị.~~*
* *~~Khi người dùng nhập mã OTP, mã sẽ được gửi lên server của RSA SecurID để kiểm tr và gửi lại phàn hồi.~~*

**KẾT LUẬN:** Em tìm hiểu tích hợp GG Authen, SMS, LoginTC theo mã OTP dùng TOTP, có thể kết hợp thêm dùng Gmail.