TT	Mục	Nội dung sửa đổi	Ngày hiệu lực
1.		Tạo mới	30/07/2015
2.		Ban hành lần 2	25/01/2017
3.	1	Bổ sung chính sách mật khẩu mạnh tại mục 1.2	25/01/2017
4.	1	Thêm danh sách mật khẩu blacklist tại phụ lục 01	25/01/2017
5.	1	Mục 1.4 bổ sung trường hợp reset mật khẩu qua tin nhắn và yêu cầu đổi mật khẩu ngay lần đầu đăng nhập	25/01/2017
6.	1	Mục 1.5 bổ sung yêu cầu về lưu trữ mã PIN	25/01/2017
7.	1	Bổ sung mục 1.10, 1.11 đưa ra tiêu chuẩn cho việc tích hợp nhận diện thuê bao qua 3G	25/01/2017
8.	1	Bổ sung mục 1.12, bổ sung yêu cầu xác thực webservice giữa các server	25/01/2017
9.	1	Bổ sung ghi chú với các chính sách mật khẩu cho ứng dụng phát triển cho các tổ chức bên ngoài	25/01/2017
10.	6	Bổ sung mục 6.2, yêu cầu về kiểu dữ liệu Content-type trả lại tương ứng	25/01/2017
11.	9	Bổ sung mục 9.3, 9.4, 9.5, yêu cầu cho các giao dịch thương mại điện tử	25/01/2017
12.	9	Bổ sung mục 9.6, 9.7 về việc yêu cầu cho mã OTP	25/01/2017
13.		Ban hành lần 3	29/01/2018
14.	1	Bổ sung ý “không phân biệt hoa thường” tại mục 1.1	29/01/2018
15.	1	Xóa ý: “đồng thời yêu cầu người dùng đổi mật khẩu ngay lần đăng nhập đầu tiên” tại mục 1.4	29/01/2018
16.	1	Bổ sung ý: “Nếu chức năng reset/quên mật khẩu sử dụng mã OTP để kiểm tra xác nhận từ người dùng, việc sử dụng mã OTP phải tuân thủ theo mục 9.7” tại mục 1.4	29/01/2018
17.	1	Chỉnh sửa mục 1.5: Bổ sung thêm thuật toán SHA-512, SHA-3	29/01/2018
18.	1	Chỉnh sửa mục 1.7: Thêm các yêu cầu về việc sinh và sử dụng Captcha	29/01/2018
19.	3	Chỉnh sửa mục 3.5: “Phải có tính năng xóa phiên...” => “Nên có tính năng xóa phiên...”	29/01/2018

20.	4	Chỉnh sửa mục 4.1: Thêm ý "sử dụng thuật toán tối thiểu là AES-256"	
21.	4	Chỉnh sửa mục 4.7.3: "bằng cách kiểm tra đồng thời file header và phần mở rộng của file" => "bằng cách kiểm tra phần mở rộng của file tương ứng với whitelist định dạng file tại mục 4.7.2"	29/01/2018
22.	4	Bổ sung ý 4.8.2	29/01/2018
23.	4	Bổ sung ý 4.10	29/01/2018
24.	5	Viết lại rõ ý 5.3	29/01/2018
25.	6	Bỏ ví dụ: "Đầu ra là json, thực hiện encode dữ liệu trả về dạng object, không trả về dạng mảng."	29/01/2018
26.	6	Chỉnh sửa mục 6.6: danh sách whitelist các URI => danh sách whitelist các địa chỉ server	29/01/2018
27.	1	Bổ sung mục 1.2	16/02/2019
28.	1	Chỉnh sửa mục 1.8, thêm yêu cầu về captcha an toàn	16/02/2019
29.	4	Bổ sung ý cho mục 4.1: masking và mã hóa dữ liệu khi trao đổi	16/02/2019
30.	8	Bổ sung ý cho mục 8.2: tải về từ nguồn chính thức hoặc đáng tin cậy	16/02/2019
31.	9	Bổ sung ý cho mục 9.7: Sinh và gửi OTP an toàn	16/02/2019
32.	1	Chỉnh sửa mục 1.9, yêu cầu bắt buộc sử dụng HTTPS	14/03/2019
33.	1	Chỉnh sửa mục 1.3, chính sách mật khẩu tuân thủ theo quy định có sẵn	14/03/2019
34.		Cập nhật phạm vi và đối tượng áp dụng	20/03/2019
35.		Ban hành lần 4	27/03/2019

	Biên soạn	Kiểm tra	Phê duyệt
Chữ ký	Vũ Thị Mai Anh		

BẢN TÓM TẮT

Tiêu chuẩn đưa ra các yêu cầu trong việc lập trình để đảm bảo ATTT cho ứng dụng web. Các đề mục lớn bao gồm:

- Yêu cầu về quản lý xác thực cần đảm bảo các yêu cầu về việc sinh, lưu trữ, quản lý các thông tin định danh và các chức năng xử lý xác thực giữa người dùng - ứng dụng, ứng dụng - ứng dụng.
- Yêu cầu về việc sinh, lưu trữ, quản lý phiên đăng nhập.
- Yêu cầu về việc phân quyền và kiểm tra quyền của các tác nhân khi tương tác với ứng dụng.
- Các yêu cầu khi tương tác với back-end:
  - Mã hóa dữ liệu nhạy cảm trước khi lưu trữ.
  - Thực hiện xác thực và truy vấn an toàn SQL
  - Thực hiện xác thực và truy vấn an toàn NoSQL.
  - Thiết lập whitelist hoặc blacklist khi tương tác với Xpath, LDAP.
  - Sử dụng các API hoặc chuẩn hóa dữ liệu từ người dùng khi tương tác với OS.
  - Xử lý download, upload file an toàn.
  - Tạo HTTP request phía server an toàn.
  - Tương tác an toàn với dữ liệu XML.
  - Tương tác an toàn với các hàm deserialize.
- Yêu cầu về việc kiểm soát dữ liệu đầu vào.
- Yêu cầu về việc kiểm soát dữ liệu đầu ra.
- Yêu cầu về việc kiểm soát ngoại lệ và ghi log ứng dụng.
- Yêu cầu khi sử dụng framework, thư viện.
- Yêu cầu về việc xử lý nghiệp vụ hệ thống (bussiness logic).

Trên đây là Bản tóm tắt Tiêu chuẩn ATTT ứng dụng web để các cơ quan, đơn vị theo dõi, thực hiện./.

I. Mục đích

- Xây dựng tiêu chuẩn ATTT cho ứng dụng web, nhằm đưa ra các yêu cầu để phát triển ứng dụng web đảm bảo an toàn thông tin.

II. Đối tượng áp dụng

- Tiêu chuẩn này áp dụng xuyên suốt với Khối cơ quan tập đoàn, các Tổng Công ty, Công ty, Trung tâm, Viện nghiên cứu, Học viện Viettel, các đơn vị hạch toán phụ thuộc khác của Tập đoàn và các công ty hạch toán độc lập do Tập đoàn sở hữu từ 50% vốn điều lệ trở lên.
- Tổng công ty VTT căn cứ theo Hợp đồng dịch vụ (SLA) với VTG thực hiện hướng dẫn các Công ty thị trường xây dựng và ban hành áp dụng tiêu chuẩn này.
- Trong vòng 45 ngày kể từ ngày tiêu chuẩn này có hiệu lực, các đơn vị có trách nhiệm ban hành các hướng dẫn cụ thể cho các framework sử dụng tại đơn vị phải đảm bảo tuân thủ tiêu chuẩn này *(Nội dung không được trái với Tiêu chuẩn này)*.

  Sau khi ban hành, các đơn vị nêu trên phải gửi một bản về công ty An ninh mạng Viettel để theo dõi, quản lý.

III. Phạm vi áp dụng

- Tiêu chuẩn này áp dụng chung cho tất cả các ứng dụng web được phát triển và sử dụng trong Tập đoàn.

IV. Các yêu cầu

1. Quản lý xác thực

➢ ***Thông tin định danh***

1.1. Tên đăng nhập phải là duy nhất, không phân biệt hoa thường, chỉ nên chứa tập các ký tự là chữ cái, chữ số, dấu gạch dưới.

1.2. Không sử dụng chung thông tin đăng nhập:

– Nếu trong một hệ thống dịch vụ có nhiều thông tin xác thực giữa các tiến trình, dịch vụ thì thông tin xác thực phải khác nhau hoàn toàn.

1.3. Thiết lập chính sách mật khẩu mạnh:

– Tuân thủ theo chính sách mật khẩu tại Quyết định số 2322A/QĐ-CNVTQĐ-CNTT.

– Các mật khẩu đã đảm bảo theo chính sách trên phải nằm ngoài danh sách blacklist các mật khẩu thông dụng, chi tiết xem tại Phụ lục 01.

1.4. Thiết lập thời gian hết hiệu lực cho mật khẩu tối đa 90 ngày, mật khẩu mới không được trùng với mật khẩu hiện tại.

1.5. Đối với chức năng reset/ quên mật khẩu:

- Đường dẫn reset/quên mật khẩu được gửi qua email phải bị mất hiệu lực sau lần truy cập đầu tiên hoặc sau 8 giờ nếu không được truy cập.
- Nếu chức năng reset/quên mật khẩu thực hiện gửi mật khẩu qua email, tin nhắn thì mật khẩu phải được sinh ngẫu nhiên và phải tuân theo chính sách mật khẩu mạnh tại mục 1.3.
- Nếu chức năng reset/quên mật khẩu sử dụng mã OTP để kiểm tra xác nhận từ người dùng, việc sử dụng mã OTP phải tuân thủ theo mục 9.7.

1.6. Chỉ lưu dạng mã hash của mật khẩu, mã PIN trong database (DB), sử dụng thuật toán hash từ SHA-256, SHA-512, SHA-3 và các thuật toán tương đương.

➢ ***Xử lý xác thực***

1.7. Trả về thông báo chung cho trường hợp người dùng đăng ký thông tin định danh (username, email,…) đã tồn tại tại chức năng đăng ký, hoặc gửi sai thông tin định danh tại các chức năng đăng nhập, reset/quên mật khẩu, đổi địa chỉ email,…

1.8. Bật cơ chế bảo vệ bằng Captcha hoặc các hình thức tương đương khi đăng nhập sai quá 5 lần liên tiếp. Phải triển khai cơ chế này tại các chức năng quan trọng khác của ứng dụng. Captcha phải thỏa mãn các điều kiện sau:

- Sử dụng reCaptcha nếu phù hợp với nghiệp vụ và không ảnh hưởng đến các vấn đề khác.
- Nếu tự động sinh captcha thì captcha phải đáp ứng được: background (nền) phải có màu thay đổi, font chữ của captcha phải thay đổi, có độ nghiêng, có làm nhiễu, thay đổi vị trí của chữ ngẫu nhiên,..
- Captcha phải có giá trị ngẫu nhiên, bao gồm chữ cái và chữ số, độ dài tối thiểu là 5 ký tự.
- Việc đếm số lần đăng nhập sai không được dựa theo SessionID và giá trị đếm số lần đăng nhập sai không được lưu phía client.

1.9. Chỉ sử dụng phương thức POST để submit thông tin định danh, bắt buộc sử dụng HTTPS cho đường truyền để tăng tính bảo mật.

1.10. Tắt tính năng "autocomplete" đối với form chứa thông tin nhạy cảm như: mã thẻ ngân hàng, mã thẻ cào, mã thẻ khách hàng…

1.11. Với các ứng dụng xác thực người dùng qua nhận diện thuê bao, bắt buộc phải truy vấn thông tin qua hệ thống VAAA.

1.12. Với webservice dùng cho ứng dụng mobile có sử dụng cơ chế nhận diện thuê bao, phải có cảnh báo người dùng với 2 cấp độ sau:

- Cấp độ 1: Bắt buộc với các ứng dụng chứa dữ liệu cá nhân của người dùng như họ tên, CMTND, địa chỉ, ngày sinh, thông tin cước, liên hệ, nghề nghiệp, chức vụ… Yêu cầu xác nhận qua USSD hoặc nhập Captcha với nội dung ảnh Captcha có chứa thông tin về ứng dụng, đồng thời nhắn tin cảnh báo người dùng mỗi lần đăng nhập thành công.
- Cấp độ 2: Áp dụng đối với các ứng dụng không nằm ở cấp độ 1. Yêu cầu có tin nhắn cảnh báo người dùng mỗi lần đăng nhập thành công.

1.13. Đối với các webservice thuộc mô hình server gọi server, việc xác thực phải đảm bảo ít nhất 3 yếu tố: username, password và ip của server.

Ghi chú: Đối với ứng dụng phát triển cho các tổ chức bên ngoài, các yêu cầu tại mục 1.3, 1.4 có thể áp dụng chính sách theo yêu cầu của khách hàng.

2. Quản lý phiên đăng nhập

2.1. Session phải được quản lý bởi server, sinh ngẫu nhiên và độ dài tối thiểu là 128-bit.

2.2. Session phải được thiết lập thời gian timeout, giá trị timeout nên cân bằng giữa nhu cầu thương mại và yếu tố bảo mật.

2.3. Tạo mới session sau khi đăng nhập thành công.

2.4. Xóa giá trị sessionid và các dữ liệu gắn với session đó khi người dùng đăng xuất.

2.5. Cấu hình thuộc tính "Secure" đối với các ứng dụng sử dụng HTTPS và "HTTP-Only" cho trường Cookie.

2.6. Đối với các chức năng quan trọng có tương tác với database, ứng với mỗi phiên phải sinh thêm 1 token ngẫu nhiên, và thực hiện kiểm tra tính hợp lệ của token này trước khi xử lý truy vấn từ người dùng.

3. Phân quyền

3.1. Kiểm tra phân quyền dựa trên các đối tượng được lưu tại server (ví dụ: tham số lưu trên session server, dữ liệu lưu trên DB,…).

3.2. Phân quyền tối thiểu, chỉ đáp ứng đủ chức năng và tài nguyên cho người dùng/ứng dụng.

3.3. Phía giao diện người dùng: Chỉ hiển thị các thành phần giao diện, đường dẫn, hàm,... tương ứng với quyền của người dùng.

3.4. Phía server: Kiểm tra quyền tác động của người dùng/ứng dụng trên các hàm và tài nguyên tương ứng trước khi thực hiện bất cứ tác vụ nào tới hệ thống.

3.5. Nên có tính năng xóa phiên làm việc hiện tại của người dùng hoặc các cơ chế tương đương đối với các trường hợp quyền người dùng bị thay đổi hoặc bị disable bởi người dùng có thẩm quyền.

3.6. Không đặt trang quản trị public internet, trong trường hợp bắt buộc phải đặt public phải giới hạn các IP được phép truy cập hoặc sử dụng cơ chế xác thực đa nhân tố.

4. Tương tác với back-end

4.1. Mã hóa các dữ liệu nhạy cảm

Đối với các loại dữ liệu nhạy cảm như thông tin tài khoản ngân hàng, private key… phải thực hiện mã hóa trước khi lưu trữ, sử dụng thuật toán AES-256 hoặc các thuật toán tương đương.

Đối với các loại dữ liệu nhạy cảm như thông tin cá nhân, tài khoản ngân hàng,... phải được mã hóa hoặc masking khi trao đổi phù hợp với nghiệp vụ.

4.2. SQL

4.2.1. Sử dụng mô hình truy vấn prepared statement (parameterized query) hoặc các hình thức tương đương.

4.2.2. Trong 1 số trường hợp không sử dụng được các mô hình ở trên, phải thiết lập danh sách whitelist các đầu vào mong muốn.

4.3. NoSQL

4.3.1. Không công khai dịch vụ ra mạng internet, cài đặt trong môi trường mạng an toàn.

4.3.2. Đối với các hệ NoSQL có hỗ trợ xác thực, phải cấu hình xác thực khi truy cập.

4.3.3. Phụ thuộc vào hệ NoSQL sử dụng, sử dụng các api hỗ trợ truy vấn an toàn hoặc thực hiện escape các ký tự đặc biệt khi xây dựng câu truy vấn.

4.4. XPath

4.4.1. Thiết lập danh sách whitelist các ký tự đầu vào mong muốn, đầu vào nên là tập hợp của chữ cái, chữ số.

4.4.2. Lập danh sách blacklist các ký tự đặc biệt (( ) = ‘ [ ] : , * / và dấu cách), loại bỏ các đầu vào có chứa các ký tự nằm trong blacklist.

4.5. LDAP

4.5.1. Thiết lập danh sách whitelist các ký tự đầu vào mong muốn, đầu vào nên là tập hợp của chữ cái, chữ số.

4.5.2. Lập danh sách blacklist các ký tự đặc biệt (( ) ; , * | & = và nullbyte), loại bỏ các đầu vào có chứa các ký tự nằm trong blacklist.

4.6. Tương tác với OS

4.6.1. Sử dụng các API hỗ trợ việc thực thi câu lệnh hệ thống.

4.6.2. Không truyền trực tiếp dữ liệu người dùng truyền lên tới OS, trong trường hợp bắt buộc phải thiết lập danh sách whitelist các đầu vào mong muốn.

4.7. Tương tác với file

4.7.1. Không truyền trực tiếp dữ liệu từ người dùng đến các hàm include file.

4.7.2. Lập danh sách whitelist các định dạng file được phép upload.

4.7.3. Validate file hợp lệ bằng cách kiểm tra phần mở rộng của file tương ứng với whitelist định dạng file tại mục 4.7.2.

4.7.4. Với các trường hợp không bắt buộc thì không lưu file upload trong thư mục web, bỏ quyền thực thi trên thư mục upload.

4.7.5. Khi cần ánh xạ tới các file tồn tại trên hệ thống phải thiết lập danh sách whitelist đầu vào mong muốn hoặc gán các giá trị định danh tương ứng file thay vì truyền tên file.

4.7.6. Không trả về đường dẫn tuyệt đối của file.

4.7.7. Tất cả dữ liệu, tài nguyên hệ thống (báo cáo, file upload, file cấu hình...) không được lưu trong thư mục cho phép truy cập trực tiếp không qua xác thực.

4.8. Xử lý back-end HTTP request

4.8.1. Khi tạo HTTP request phía server, các tham số GET, POST cho request đó tránh tạo từ dữ liệu phía người dùng, hoặc phải được kiểm tra cẩn thận để chống ghi đè các tham số khác.

4.8.2. Không lấy địa chỉ server từ dữ liệu người dùng gửi lên. Trong trường hợp địa chỉ server cần lấy từ người dùng, phải blacklist các IP trong dải nội bộ sau khi đã phân giải DNS.

4.9. Tương tác với XML

4.9.1. Tắt tính năng "external entity resolves" và "remote doctype retrival" của xml parser khi đọc dữ liệu xml.

4.9.2. Kiểm tra dữ liệu người dùng, encode các kí tự đặc biệt (< > ' ") khi tạo dữ liệu xml.

4.10. Tương tác với các hàm deserialize

4.10.1. Khuyến nghị chỉ thực hiện deserialize các dữ liệu từ các nguồn tin cậy, an toàn hoặc sử dụng kiểu dữ liệu json.

4.10.2. Các trường hợp nằm ngoài mục 4.10.1 phải thực hiện thêm 1 trong 2 tác vụ sau:

- Sinh 1 mã bí mật (S) và lưu tại server. Khi cần gửi dữ liệu đã được serialize (D), gửi kèm mã hash được tính theo công thức: H = hash(D+S).

  Khi cần deserialize dữ liệu D, thực hiện sinh mã H1 = hash(D+S), nếu H và H1 trùng khớp mới thực hiện deserialize D.
- Thiết lập whitelist các class được deserialize. Kiểm tra tên các class trong phần dữ liệu, nếu các class này thuộc whitelist mới thực hiện deserialize dữ liệu.

5. Kiểm soát dữ liệu đầu vào

5.1. Việc kiểm tra dữ liệu đầu vào phải được thực hiện phía server.

5.2. Thực hiện việc kiểm tra dữ liệu từ tất cả các nguồn dữ liệu có tương tác với người dùng (Các tham số lấy từ GET/POST request, HTTP Headers,

dữ liệu lấy từ DB, dữ liệu từ file upload,…).

5.3. Xác định 1 kiểu encoding nhất quán sử dụng khi hiển thị, trao đổi hay lưu trữ dữ liệu. Chỉ thực hiện filter, validate dữ liệu sau khi đã đưa dữ liệu về kiểu encoding đã xác định trước đó.

5.4. Validate kiểu dữ liệu, phạm vi, kích thước dữ liệu và định dạng dữ liệu.

5.5. Nếu dữ liệu đầu vào bắt buộc là các ký tự đặc biệt, phải thiếp lập danh sách whitelist các ký tự đầu vào mong muốn.

6. Kiểm soát dữ liệu đầu ra

6.1. Phải chỉ rõ character encoding cho dữ liệu đầu ra.

6.2. Phải thiết lập giá trị Content-type tương ứng với định dạng dữ liệu trả về (ví dụ dữ liệu json phải tương ứng với Content-type là application/json)

6.3. Response body phải được encode theo ngữ cảnh sử dụng. Ví dụ: Đầu ra là html, thực hiện html encode các kí tự đặc biệt (<>’”&) từ các nguồn dữ liệu không an toàn (Các tham số lấy từ GET/POST request, HTTP Headers, dữ liệu lấy từ DB, dữ liệu từ file upload,… có thể điều khiển được bởi người dùng).

6.4. Response header: lọc bỏ các kí tự đặc biệt (\n, \r) do dữ liệu người dùng truyền vào.

6.5. Cookie trả về phải giới hạn tối thiểu nhất các thuộc tính (domain, path, httponly, expire, secure). Tránh lưu trữ các dữ liệu nhạy cảm trên cookie, nếu cần lưu trữ các dữ liệu nhạy cảm thì phải thực hiện mã hóa các dữ liệu này với thuật toán đối xứng mạnh và key chỉ được lưu trên server.

6.6. Hạn chế việc chuyển hướng, chuyển tiếp đến các URI khác. Nếu ứng dụng có chức năng này phải lập danh sách whitelist các địa chỉ server được phép thực hiện chuyển hướng, chuyển tiếp.

7. Kiểm soát ngoại lệ và ghi log ứng dụng

7.1. Xử lý các ngoại lệ bằng try-catch và trả về các thông báo lỗi chung, thông báo lỗi trả về không được chứa các thông tin nhạy cảm của người dùng, hệ thống,...

7.2. Các thông tin lỗi, ngoại lệ này phải được log lại để phục vụ bảo trì, xác định nguyên nhân lỗi ứng dụng.

7.3. File log phải được đặt tại thư mục an toàn ngoài thư mục web.

7.4. Không log lại các dữ liệu nhạy cảm (thông tin người dùng, session id, thông tin hệ thống).

7.5. Giới hạn người dùng cho phép truy cập file log.

8. Sử dụng framework, thư viện (third-party components)

8.1. Loại các code thừa, các thành phần và thư viện không cần thiết.

8.2. Sử dụng phiên bản mới nhất của framework, thư viện tại thời điểm phát triển ứng dụng và được tải về từ nguồn chính thức hoặc đáng tin cậy.

8.3. Thường xuyên cập nhật các bản vá lỗi cho framework, thư viện.

8.4. Tắt chế độ development của framework khi triển khai ứng dụng thực tế.

9. Xử lý bussiness logic

Xử lý business logic phụ vào từng ứng dụng nhưng yêu cầu:

9.1. Lập trình viên phải nắm rõ được toàn bộ luồng nghiệp vụ của ứng dụng, phải xác định các ngoại lệ cho từng nghiệp vụ để tránh các lỗi logic có thể xảy ra.

9.2. Các chức năng quan trọng (ví dụ chuyển khoản ngân hàng), sử dụng các hình thức khóa hoặc các hình thức tương đương để tránh lỗi race condition.

9.3. Với các ứng dụng sử dụng tiền ảo, tiền điện tử, các giao dịch thực hiện trừ tiền trong tài khoản người dùng chỉ được phép thực hiện sau khi hệ thống đã xác thực người dùng thành công qua tối thiểu 2 bước, ví dụ: mật khẩu + SMS OTP, mật khẩu + USSD, …

9.4. Đối với các dịch vụ viễn thông, khi khách hàng đăng ký các dịch vụ VAS phải có tin nhắn thông báo tới khách hàng.

9.5. Đối với các giao dịch chuyển tiền, ví dụ chuyển từ tài khoản A sang tài khoản B: phải thực hiện trừ tiền tài khoản A thành công rồi mới được thực hiện cộng tiền vào tài khoản B.

9.6. Đối với các ứng dụng có chức năng gửi tin nhắn tới người dùng phải giới hạn số lần gửi tin trong 1 ngày ứng với mỗi đầu số nhận tin. Đối với chức năng quan trọng như đăng ký, lấy lại mật khẩu chỉ cho phép gửi <= 3 tin/ngày.

9.7. Yêu cầu khi sử dụng và sinh mã OTP:

- Giới hạn số lần nhập sai với mỗi mã OTP <= 3 lần/ngày, xóa mã cũ

và sinh mã mới khi nhập sai vượt quá số lần cho phép.

- Không được sử dụng mã OTP làm mật khẩu.
- Đảm bảo mã OTP không thể lấy được khi có thông tin đăng nhập tài khoản.(Gửi OTP qua SMS hoặc phương thức tương đương – tách biệt với phương thức xác thực)

Nơi nhận:
- Ban TGĐ TĐ (để b/c);
- Như điều II (để t/h);
- Lưu: VT, ANM; Anh 02.

PHỤ LỤC 01: DANH SÁCH BLACKLIST MẬT KHẨU

STT	Giá trị
1	111111a@
2	111111A@
3	111111@a
4	111111@A
5	123456a@
6	123456A@
7	123456@a
8	123456@A
9	admin@123
10	admin@1234
11	admin@2010
12	admin@2011
13	admin@2012
14	admin@2013
15	admin@2014
16	admin@2015
17	admin@2016
18	admin@2017
19	Admin@123
20	Admin@1234
21	Admin@2010
22	Admin@2011
23	Admin@2012
24	Admin@2013
25	Admin@2014
26	Admin@2015
27	Admin@2016
28	Admin@2017
29	viettel@123
30	viettel@1234
31	viettel@2010
32	viettel@2011
33	viettel@2012
34	viettel@2013
35	viettel@2014
36	viettel@2015
37	viettel@2016
38	viettel@2017
39	Viettel@123

40	Viettel@1234
41	Viettel@2010
42	Viettel@2011
43	Viettel@2012
44	Viettel@2013
45	Viettel@2014
46	Viettel@2015
47	Viettel@2016
48	Viettel@2017
49	password@123
50	password@1234
51	password@2010
52	password@2011
53	password@2012
54	password@2013
55	password@2014
56	password@2015
57	password@2016
58	password@2017
59	Password@123
60	Password@1234
61	Password@2010
62	Password@2011
63	Password@2012
64	Password@2013
65	Password@2014
66	Password@2015
67	Password@2016
68	Password@2017
69	passw0rda@
70	passw0rdA@
71	passw0rd@123
72	passw0rd@1234
73	passw0rd@2010
74	passw0rd@2011
75	passw0rd@2012
76	passw0rd@2013
77	passw0rd@2014
78	passw0rd@2015
79	passw0rd@2016
80	passw0rd@2017

81	passw0rd@a
82	passw0rd@A
83	qwerty@123
84	qwerty@1234
85	qwerty@2010
86	qwerty@2011
87	qwerty@2012
88	qwerty@2013
89	qwerty@2014
90	qwerty@2015
91	qwerty@2016
92	qwerty@2017
93	abc123a@
94	abc123A@
95	abc123@123
96	abc123@1234
97	abc123@2010
98	abc123@2011
99	abc123@2012
100	abc123@2013
101	abc123@2014
102	abc123@2015
103	abc123@2016
104	abc123@2017
105	abc123@a
106	abc123@A
107	123123a@
108	123123A@
109	123123@a
110	123123@A
111	root@123
112	root@1234
113	root@2010
114	root@2011
115	root@2012
116	root@2013
117	root@2014
118	root@2015
119	root@2016
120	root@2017
121	Root@123

122	Root@1234
123	Root@2010
124	Root@2011
125	Root@2012
126	Root@2013
127	Root@2014
128	Root@2015
129	Root@2016
130	Root@2017
131	r00t@123
132	r00t@1234
133	r00t@2010
134	r00t@2011
135	r00t@2012
136	r00t@2013
137	r00t@2014
138	r00t@2015
139	r00t@2016
140	r00t@2017
141	R00t@123
142	R00t@1234
143	R00t@2010
144	R00t@2011
145	R00t@2012
146	R00t@2013
147	R00t@2014
148	R00t@2015
149	R00t@2016
150	R00t@2017
151	qazwsx@123
152	qazwsx@1234
153	qazwsx@2010
154	qazwsx@2011
155	qazwsx@2012
156	qazwsx@2013
157	qazwsx@2014
158	qazwsx@2015
159	qazwsx@2016
160	qazwsx@2017
161	123qwea@
162	123qweA@

163	123qwe@123
164	123qwe@1234
165	123qwe@2010
166	123qwe@2011
167	123qwe@2012
168	123qwe@2013
169	123qwe@2014
170	123qwe@2015
171	123qwe@2016
172	123qwe@2017
173	123qwe@a
174	123qwe@A