이중토큰 인증기술 및 활용
Dual Token Authentication and Their Applications

2019. 5.

중부대학교 정보보호학과
이 병 천 교수
sultan@joongbu.ac.kr

Outline

1. 이중토큰 인증기술
   - 초기인증 vs. 인증유지
   - 상태형 인증 vs. 무상태형 인증
   - 이중토큰 인증기술

2. 활용 분야
   2.1 웹 인증
   2.2 Wi-Fi 보안 프로토콜
   2.3 사물인터넷 인증
   2.4 가상데스크톱 환경의 인증

패스워드 기반 인증의 취약성

- 세상 모든 곳에 존재하는 패스워드 인증
  - 취약한 패스워드 의존을 줄이자.
  - 보안성, 효율성, 편의성 고려

- 패스워드 없는 사용자 인증 추구

초기인증 vs. 인증유지

클라이언트

처음 접속시의 사용자 신분 인증
ID/pass, 인증서, Biometrics, 멀티팩터 인증 등 다양한 인증 사용
서버측에서는 사용자 계정 DB 확인 등 엄밀한 검증 절차 필요

초기인증기술 (상태형)

인증유지기술 (무상태형)

서버

초기 인증 완료된 사용자의 인증된 상태를 오랜기간 유지하는 기술
서버가 사용자 정보를 관리할 필요 없는 무상태 서비스(stateless service) 요구
쿠키, 세션, 토큰 등의 기술 사용

사용자 편의성, 서비스 운영자의 효율성 측면에서 매우 중요

* 상태형(stateful) 인증 : 서버가 사용자의 정보를 유지해야 하는 인증기술

* 무상태형(stateless) 인증 : 서버가 사용자의 정보를 유지하지 않아도 되는 인증기술

기존의 인증유지 기술

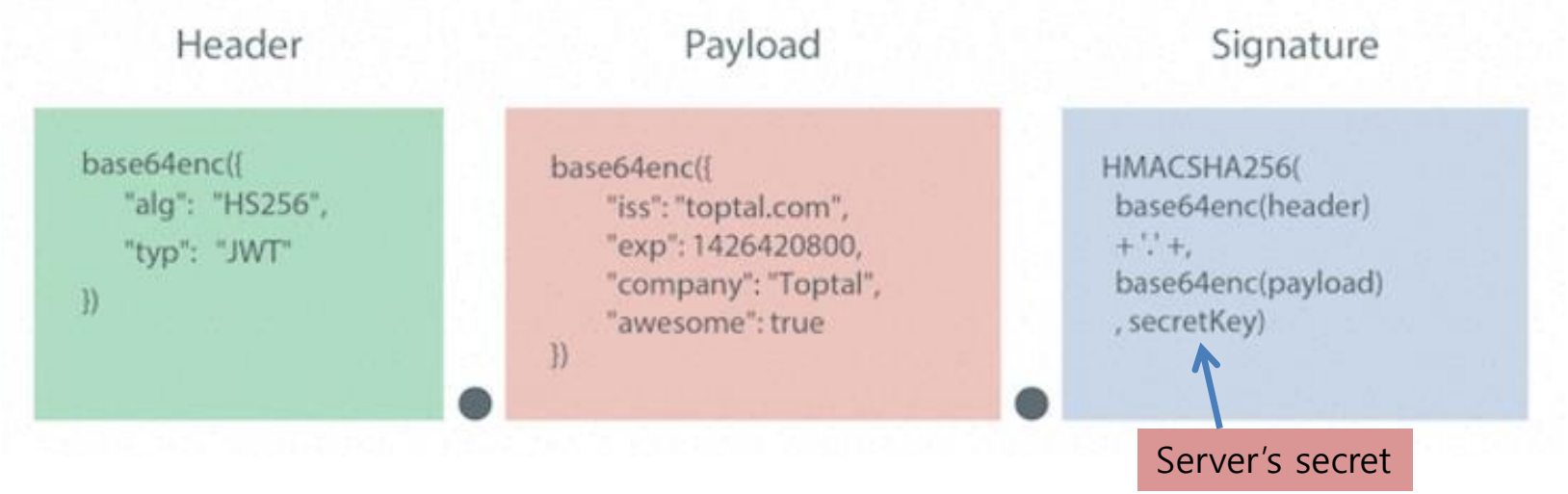
공격위협

1. 도청공격으로 쿠키를 탈취하여 ID 도용
2. 세션ID탈취 탈취하여 로그인세션 가로채기
3. 도청으로 토큰을 탈취하여 ID 도용

도청을 방지하기 위해
보안통신환경에서
운영해야 함

JSON Web Token (JWT)

- Payload에 사용자 정보를 입력하고 Signature에 HMAC 서명 부가
- 서명은 서버 비밀값을 이용한 HMAC 계산으로 서버만 계산 및 검증 가능. 제3자의 위조 불가.
- 유효기간을 가진 토큰정보



기존의 인증유지기술 비교

	쿠키	세션	토큰
정보형태	텍스트정보	랜덤한 ID 정보	서버가 서명한 정보
사용횟수	N회	1회	N회
도청시 위협	로그인 아이디 도용	로그인 세션 탈취	로그인 아이디 도용
상태/무상태	무상태	상태 (서버에 세션정보 저장)	무상태
서버 계산량	없음	세션정보 확인 필요	토큰의 서명 검증 필요
보안통신 필요성	Y	Y	Y
보안통신 통합환경의 상태/무상태	**상태**	**상태**	**상태**

HTTPS 등 보안통신을 위해서는 서버가 동시접속된 사용자의 보안세션정보를 유지해야 함

쿠키/세션/토큰 기술은 서버의 효율성을 위해 애초에 무상태 인증을 위해 고안되었지만 보안통신환경에서 사용해야 하므로 진정한 무상태 인증을 제공하지 못함

→ 평문통신환경에서도 사용 가능한 진정한 무상태 인증 기술의 필요성

이중토큰 인증기술

- 목표
  - 보안통신 기술을 사용하지 않고도
  - 토큰인증을 안전하게 수행하여
  - 무상태형 인증유지 서비스를 제공함
- 방법론
  - 서버가 서명된 이중토큰을 발급 (위조 불가)
    - 공개토큰: 서명된 아이디 역할
    - 비밀토큰: 서명된 패스워드 역할
      - 조건: 서버는 공개토큰으로부터 비밀토큰을 계산 가능한 형태
  - 이중토큰을 이용한 무상태 인증 제공
    - 서버는 자신이 발급한 토큰을 저장할 필요가 없음

이중토큰 인증기술

클라이언트

초기인증

서버

이중토큰(공개/비밀) 발급

서버 비밀키 K

$< t_p, t_s >$

클라이언트는
공개토큰과 비밀토큰을
저장하여 사용

1. 공개토큰
(서명된 아이디 역할)

2. 비밀토큰
(서명된 패스워드 역할)

사용자 정보
유효기간 포함

$t_p = HMAC(Info, K)$

$t_s = HMAC(t_p, K)$

서버는 자신이 발급한
공개토큰과 비밀토큰을
저장할 필요가 없음
(무상태 서비스 가능)

(1) 토큰은
서버만 생성 가능

(2) 공개토큰이 주어지면 서버는
언제든지 비밀토큰 생성 가능

이중토큰 인증기술

1. 이중토큰 발급

공개토큰 $t_p = HMAC(Info, K)$

비밀토큰 $t_s = HMAC(t_p, K)$

초기인증 (암호화 통신)

클라이언트

1. 이중토큰 발급 (암호화 통신)

서버

2. 이중토큰을 이용한 인증 (평문 통신)

서버 비밀키 K

2. 이중토큰을 이용한 인증

$auth = H(t_s, CurrTime)$

일회용 인증정보
현재시간 정보에 따라 바뀜

$< CurrTime,\ t_p,\ auth >$

$t_s = HMAC(t_p, K)$

$auth =? H(t_s, CurrTime)$

3. 이중토큰을 이용한 암호화 통신

$auth = H(t_s, CurrTime)$

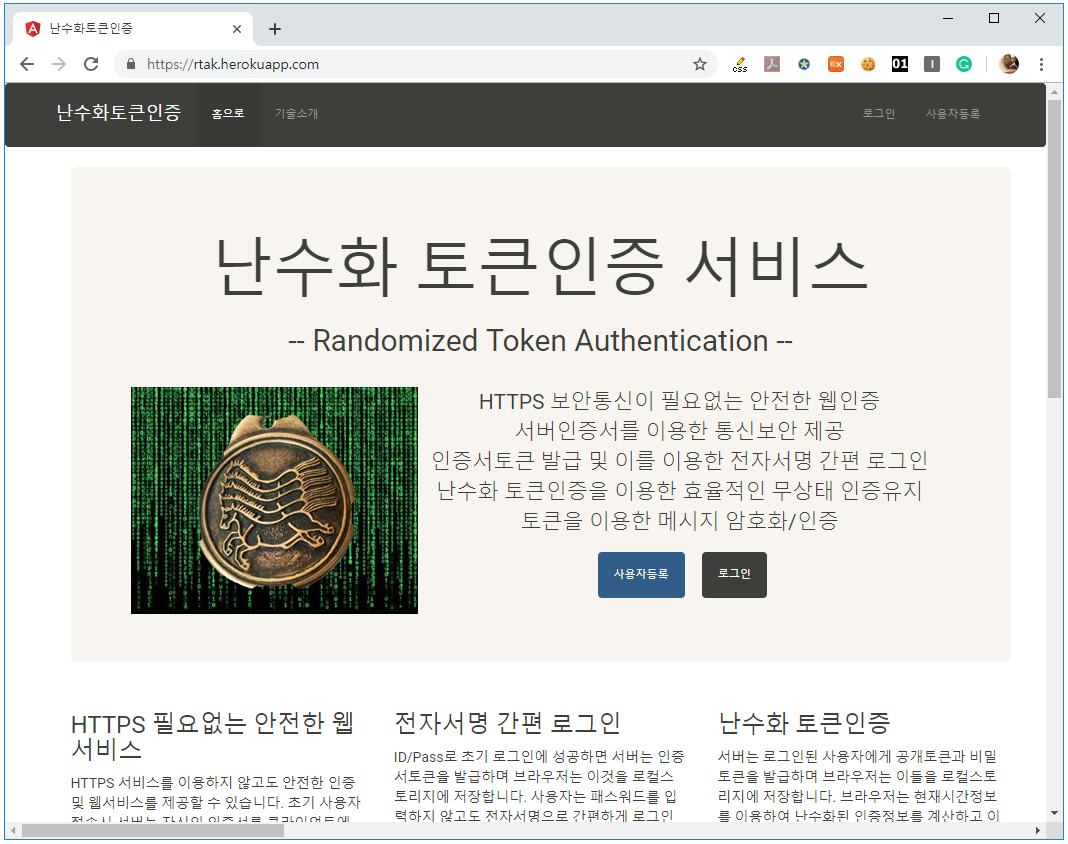
$< CurrTime,\ t_p >$

$auth = H(t_s, CurrTime)$

$auth$를 세션키로 이용한 암호화 통신

데모

- https://rtak.herokuapp.com/

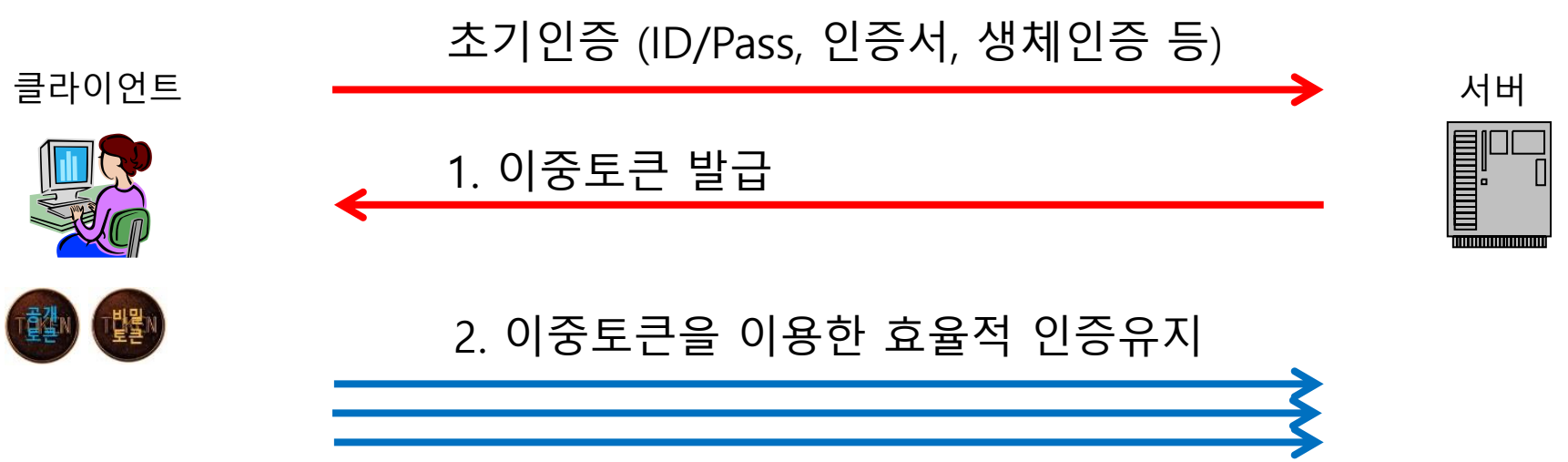


이중토큰 인증기술의 가치

- 사용자에게 새로운 인증수단 제공
  - 변조 불가 (서버가 서명한 토큰)
  - 재전송 불가 (인증값은 시간에 따라 변경됨, 도청무용)
  - 평문통신 환경에서도 안전한 인증 가능
  - 서버와의 1:1 인증유지에 사용
  - 서버는 무상태 인증 서비스 가능 (토큰 저장 불필요)
  - 사용자 개입이 필요없는 자동화된 인증에 최적
  - 해시 기반의 경량 암호기술로 구성
- 경량 암호화 통신으로 활용 가능
  - 일회용 인증값을 세션키로 활용하여 암호화 통신
- 인증분야의 새로운 원천기반기술

인증된 사용자에게 VIP 대접을 하는 올바른 방법론

적용분야 1. 웹 인증



2. 이중토큰을 이용한 인증

$auth = H(t_s, CurrTime)$

일회용 인증정보
현재시간 정보에 따라 바뀜

$< CurrTime,\ t_p,\ auth >$

$t_s = HMAC(t_p, K)$

$auth =? H(t_s, CurrTime)$

3. 이중토큰을 이용한 암호화 통신

$auth = H(t_s, CurrTime)$

$< CurrTime,\ t_p >$

$auth$를 세션키로 이용한 암호화 통신

$auth = H(t_s, CurrTime)$

Bearer 토큰과의 효율성 비교

	JWT Bearer토큰	이중토큰
정보형태	1개	2개 (공개토큰, 비밀토큰)
인증 사용횟수	N회	N회
도청시 위협	로그인 아이디 도용	안전, 재전송공격 불가
상태/무상태	무상태	무상태
서버 계산량	토큰의 서명 검증	토큰의 서명 검증 + 비밀토큰 계산 (hash) + 인증값 검증 (hash)
인증유지시 보안통신 필요성	Yes	No
보안통신 통합환경의 상태/무상태	상태	무상태
보안통신 기능	외부기술(HTTPS) 사용	자체 보안통신 가능

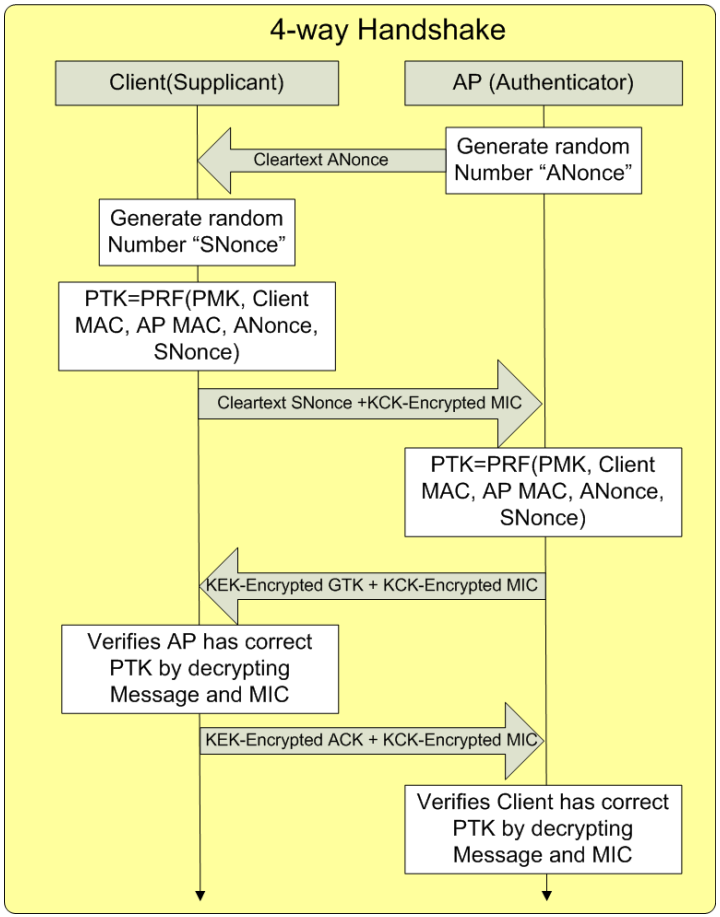
ID/Pass와 이중토큰 비교

	ID / Pass	공개토큰 / 비밀토큰
역할	ID: 사용자 정보 (공개정보) Pass: 사용자 인증 (비밀정보)	공개토큰: ID 역할 비밀토큰: 패스워드와 비슷한 역할
정보 형태	사용자가 선택하고 기억하며 사용자가 입력하는 정보	서버가 서명하여 발급한 정보 클라이언트에 저장하여 사용
정보의 관계	ID와 Pass는 상관관계 없음	서버는 공개토큰으로부터 비밀토큰을 생성할 수 있는 특수관계
통신 전달	Pass는 서버로 직접 전송되어야 함 (암호화통신 필요, 서버에 대한 신뢰 필요)	비밀토큰은 직접 전달하지 않고 비밀토큰을 이용한 계산값만을 전달 (평문통신 가능)
값의 선택 주체	사용자	서버
인증 상태	상태형 인증	무상태형 인증
정보 노출시 피해	동일한 Pass를 여러곳에 사용시 큰 피해	해당 정보는 해당 서버에서만 사용 가능
보안통신 기능 추가	외부기술(HTTPS) 사용 필요	자체 보안통신 가능

적용분야 2. Wi-Fi 보안 프로토콜

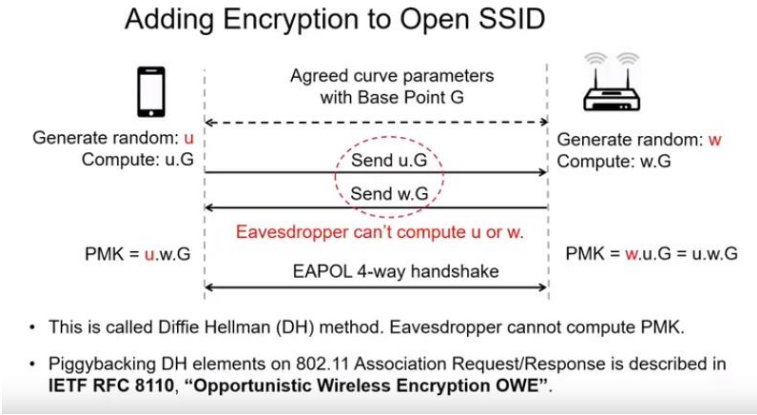
- WPA2 프로토콜
  - 공개 Wi-Fi로 운영하면 공유비밀키를 공격자와도 공유 (공격자는 사용자의 보안세션설정과정을 도청하면 세션키를 계산하여 사용자의 모든 통신을 도청 가능)
- WPA3 프로토콜
  - 모듈러승산을 이용하도록 프로토콜 개선하여 위 문제를 해결 (값비싼 해결책, 여전히 공유비밀키에 의존)
- 효율성 문제
  - 매 접속시마다 4-way 핸드세이크 수행, PSK 확인, 세션키 공유
- 이중토큰을 이용한 새로운 Wi-Fi 보안 프로토콜 제안
  - 초기인증(1회)과 보안세션설정(n회)을 구분
  - 초기 인증시 AP가 클라이언트에게 이중토큰 발급
  - 이중토큰 발급된 환경에서는 빠르게 보안세션 설정 가능

WPA2



매 접속시마다 4-way 핸드쉐이크를
반복 수행하여 세션키 생성

WPA3



DH 키합의로 생성된 PMK를 공유비밀키로 사용

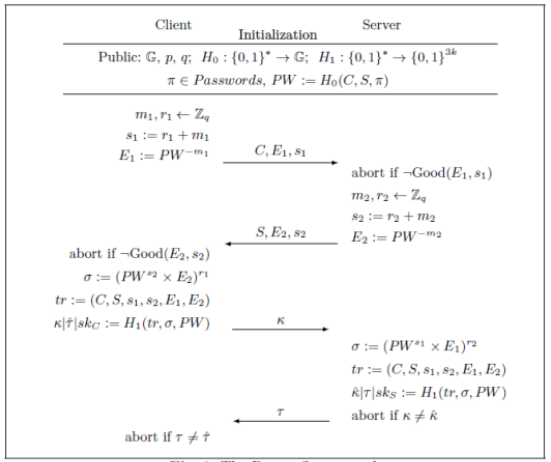


Fig. 1: The Dragonfly protocol.

모듈러승산을 이용하여 4-way 핸드쉐이크의
보안 강화

초기인증 및 이중토큰 발급

Client

Pre-shared key PSK

AP

Certificate $Cert_{AP}$
Secret key K
Pre-shared key PSK

(1) Request service

(2) Send $Cert_{AP}$

인증서를 이용한
AP 신뢰성 검증

Verify $Cert_{AP}$
Generate random key r
Compute $E_{pk}(r, pk_{AP})$

(3) Send $E_{pk}(r, pk_{AP})$

Retrieve r
Generate random challenge c

(4) Send challenge c

토큰 전송용 난수키 공유

Compute $E(PSK||c, r)$

(5) Send $E(PSK||c, r)$

Verify PSK
Generate tokens
– $t_p = HMAC(I_A, K)$
– $t_s = HMAC(T_p, K)$
– Generate T_p, T_s
– $E(T_p||T_s, r)$

(6) Send tokens $E(T_p||T_s, r)$

Retrieve T_p, T_s

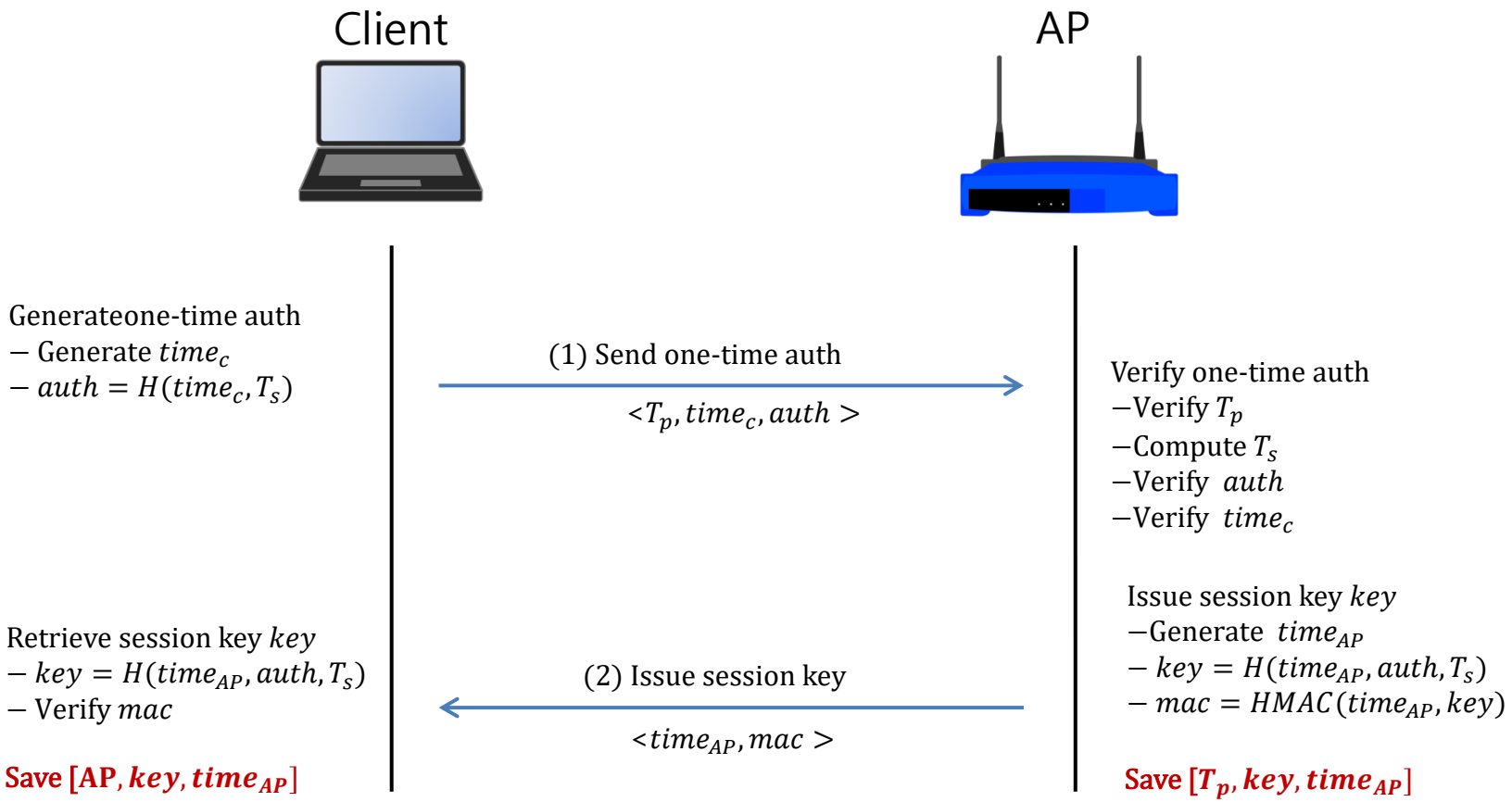
토큰을 난수키로
암호화 전송

Save [AP, T_p, T_s]

토큰 저장

토큰 저장 필요 없음

이중토큰을 이용한 보안세션 설정



1회의 평문통신으로 세션키 설정 가능

특징, 효율성

- 초기인증과 보안세션 설정의 분리
  - 1회 초기인증시 이중토큰 발급
  - n회 보안세션 설정을 효율적으로 수행: 이중토큰을 이용하여 1회의 평문통신을 주고받음으로써 보안세션 설정 완료

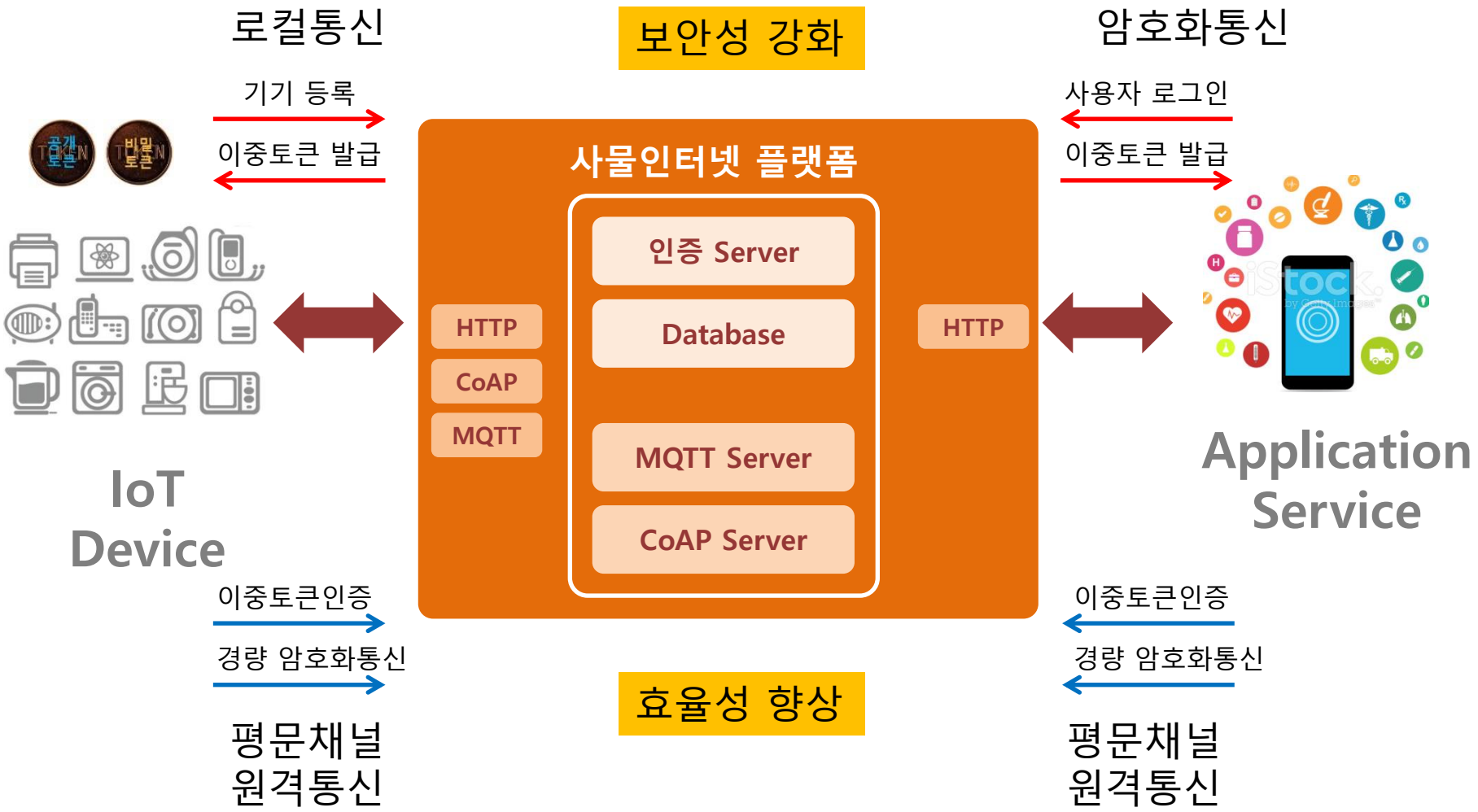
- 무상태 보안세션 설정
  - AP는 사용자의 토큰정보를 저장할 필요 없음
  - Enterprise 환경에서 매 접속시마다 별도의 인증서버를 통해 사용자 인증을 수행하는 것과 비교시 효율성 크게 향상

적용분야 3. 사물인터넷 인증

- IoT 요구조건
  - 인간의 개입 없는 자동화된 인증 필요
  - 평문통신 환경에서도 안전한 인증 필요
  - 필요시 암호화 통신 기능 제공
  - 경량 암호기술 필요
  - IoT 기기는 하나의 서버에 연결되어 사용되는 특수환경

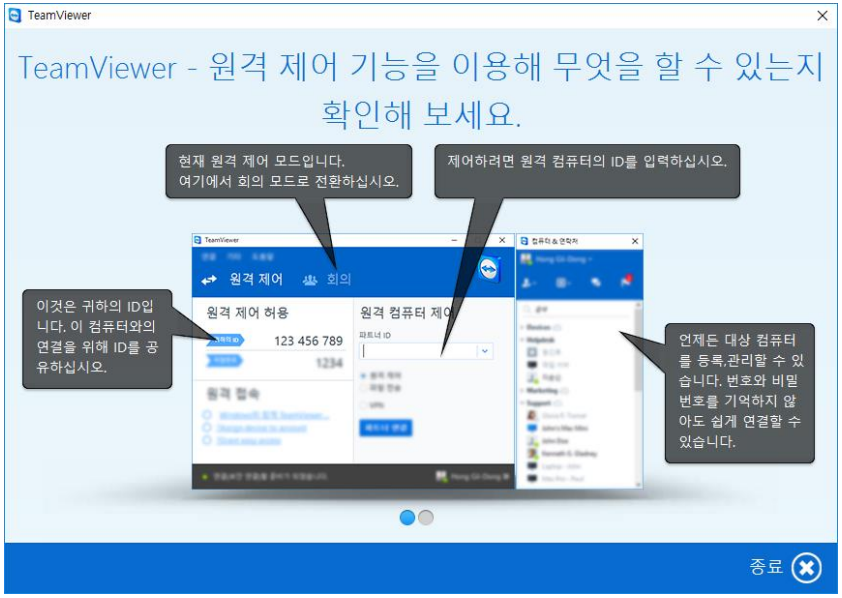
- 방법론
  - 기기 등록시 서버가 이중토큰 발급
  - 이중토큰을 이용한 안전한 인증 및 암호화 통신

사물인터넷 인증기술

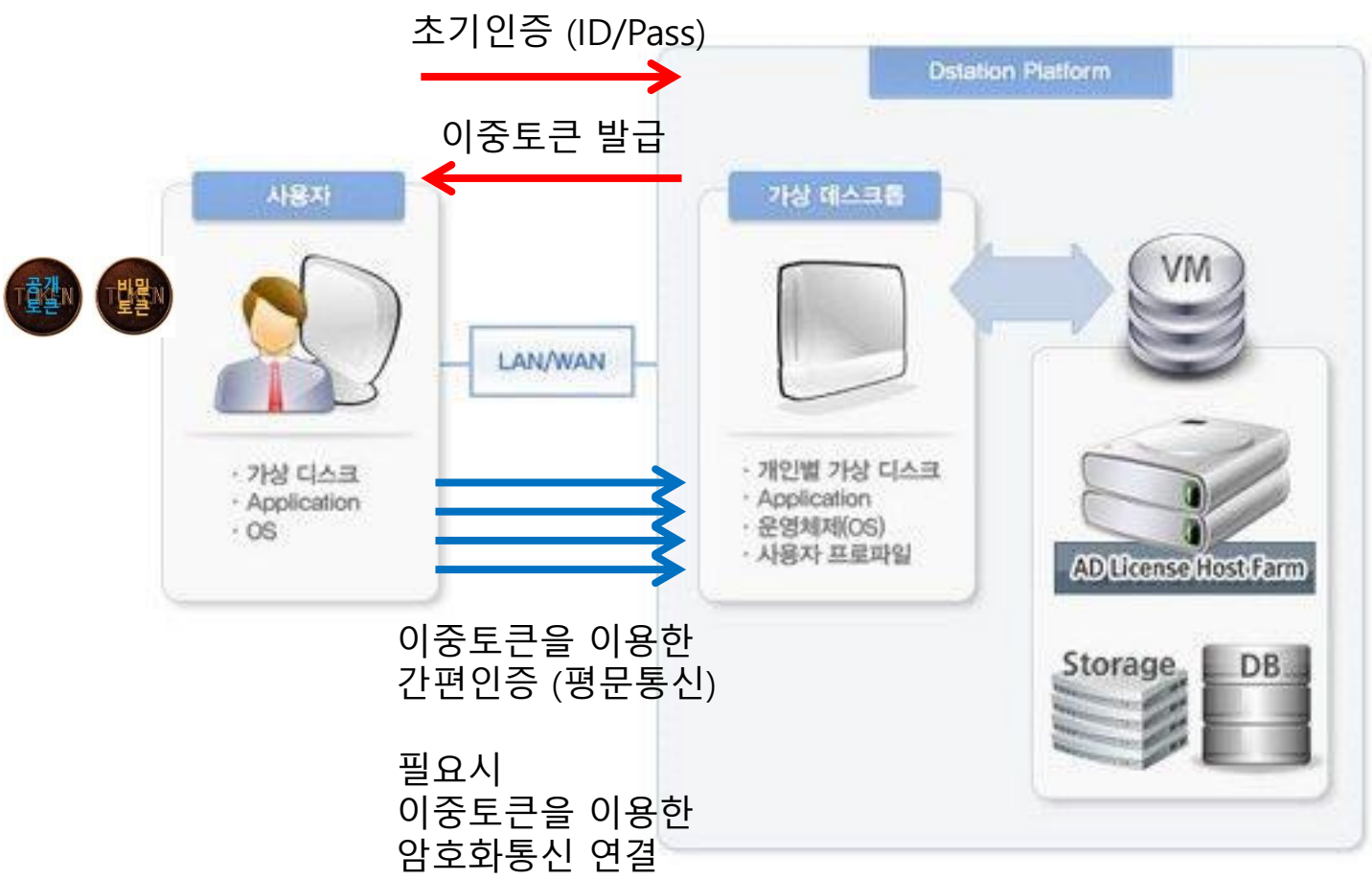

로컬통신
보안성 강화
암호화통신
기기 등록
이중토큰 발급
사용자 로그인
이중토큰 발급
사물인터넷 플랫폼
인증 Server
Database
MQTT Server
CoAP Server
HTTP
CoAP
MQTT
HTTP
IoT Device
Application Service
이중토큰인증
경량 암호화통신
이중토큰인증
경량 암호화통신
효율성 향상
평문채널 원격통신
평문채널 원격통신

적용분야 4. 가상데스크톱 인증

- 패스워드 기반 원격접속 서비스 제공?



이중토큰 기반 가상데스크톱 인증



결 론

- 이중토큰 기술은 새로운 인증 기반기술
  - 평문통신 환경에서도 안전한 무상태 인증유지 기능 제공
  - 경량 보안통신 환경 제공
- 이중토큰 인증기술은 모든 인증 환경에 적용 가능
  - 초기인증과 인증유지 기술의 병행 사용 필요
  - 사용자의 편의성 향상
  - 서버 운영자의 운영 효율성 향상, 설비 부담 감소
- 적용 확산을 위한 노력 필요
  - 선도적 연구개발, 산학협력
  - 인증, 표준화 노력