

## TTPs \$ ScarCruft Tracking Note

Archived: 2026-04-05 20:56:25 UTC

본 보고서에서 분석한 조직은 최소 2012년 부터 국내를 대상으로 공격 활동을 펼치고 있습니다. 공격 조직의 이름은 각 보안업체들 마다 명명한 여러가지 이름으로 불리고 있으며 대표적인 이름은 아래와 같습니다. 본 보고서에서는 Kaspersky로부터 공유받은 샘플로부터 추적을 시작하였기 때문에 ScarCruft라는 이름을 차용하여 설명합니다.

AKA	Named By
ScarCruft	Kaspersky
APT37	Mandiant
금성121	ESTsecurity
Red Eyes	AhnLab
Ricochet Chollima	CrowdStrike

ScarCruft 그룹은 국가기반 해킹 조직으로, 국가의 체제 유지를 위해 공격을 수행하는것으로 보입니다. 체제를 이탈한 주민, 해외 파견 근로자, 기자 및 선교사를 주 공격 대상으로 삼고 있습니다. 뿐만 아니라 공격 조직이 속한 국가와 관련된 국내 주요 인사들도 공격 대상입니다. 공격 대상에 대해 해킹에 성공하면 대상의 휴대전화 기록, 데스크톱 기록, 메신저 채팅 기록등을 탈취합니다. 침해한 시스템을 파괴하거나 탈취한 정보로 협박하는 등의 행위를 하지 않는 것으로 보아 감시 목적이 강한 것으로 추정됩니다.

목표 대상이 관심을 가질 만한 내용으로 스피어피싱 이메일을 송부하여 악성코드를 다운로드 받도록 유도하는 방식을 많이 사용합니다. 또한 메신저 채널에 위장 잠입하여 악성앱 설치를 유도하기도 합니다.

한국인터넷진흥원은 기존 발표한 보고서(TTPs #9 개인의 일상을 감시하는 공격전략 분석) 이후 지속적인 추적을 통해 ScarCruft 공격조직이 새로이 사용한 신규 명령제어 채널을 발견하였습니다. 우리는 신규 명령채널을 모니터링 할 수 있는 도구를 직접 개발 제작하여 작년 10월 중순부터 약 2개월간 공격자의 명령제어 과정을 24시간 추적하고 모니터링 하는 활동을 수행했습니다. 그 결과, 해당 조직의 타겟으로 확인된 감염자를 확인하여, 피해가 확산되지 않도록 조치했습니다. 공격자는 타겟을 감시하기 위하여 Chinotto 악

성코드를 적극적으로 활용하며 VNC 설치, 정보유출 등의 활동을 하는 것으로 확인했습니다. 그리고 공격 대상의 환경과 공격목적에 맞게 명령제어 체계를 재정비하는 정황도 확인하였습니다.

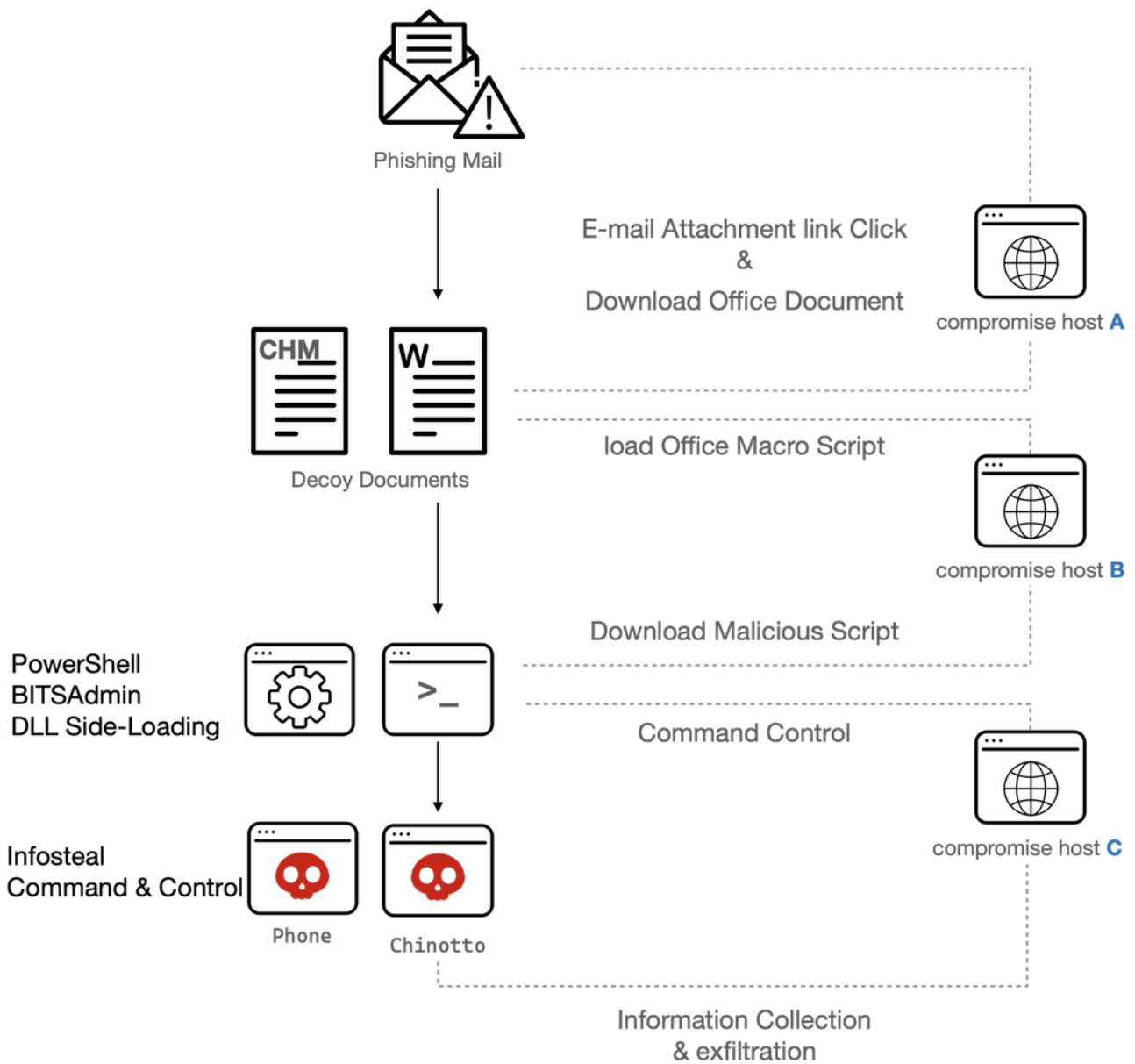
KrCert/CC는 지속적인 추적 활동을 통해 공격자의 변화된 TTP를 확인하고 이것을 알림으로써 방어자들에게 각자의 방어 환경에 맞는 전략을 수립하는데 도움을 줄 것입니다. 또한 단순히 공격자의 행위를 추적하고 TTPs를 파악하는 활동을 넘어 피해사실 인지 즉시 조치해 민간 기업과 개인의 위협을 제거하고 억지하기 위해 끊임없이 노력하고 있습니다.

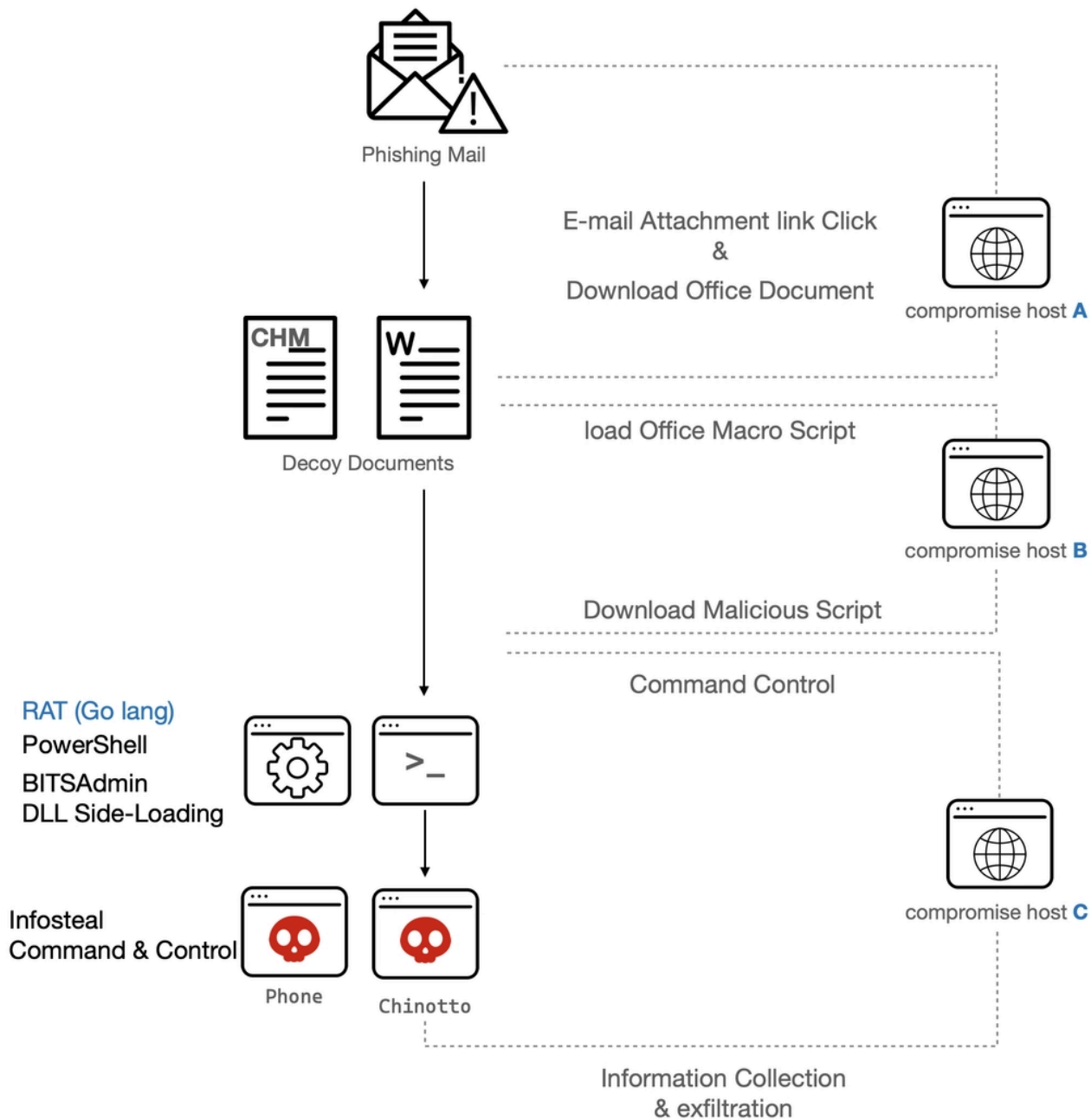
2022년 연말 공개한 TTPs #9 개인의 일상을 감시하는 공격전략 분석을 통해, 고도화된 정보수집 활동과 공격 기법, 전술, 그리고 절차에 대하여 공유하였습니다. 본 문서에서는 보고서 게시 시점에 공개하지 않았던 Go 언어 기반의 새로운 명령제어 기법에 대하여 설명하고, 악성코드에 내재된 API 키를 활용하여 공격자의 행위를 모니터링한 결과를 공유합니다.

공격자는 공격을 수행하고 지속하기 위해 TTP 중 일부를 지속적으로 변화하며 공격을 수행하고 있습니다. 우리는 위협을 추적 및 제거하고 분석하는 과정에서 기술(Techniques)의 변화를 확인했습니다. 기존 공격 과정에서는 명령제어를 위해 스크립트 기반의 명령제어를 활용하였으나, 현재 Golang 기반 명령제어 체계를 구축해 더욱 더 은밀하게 명령제어를 수행하고 있음을 확인했습니다.

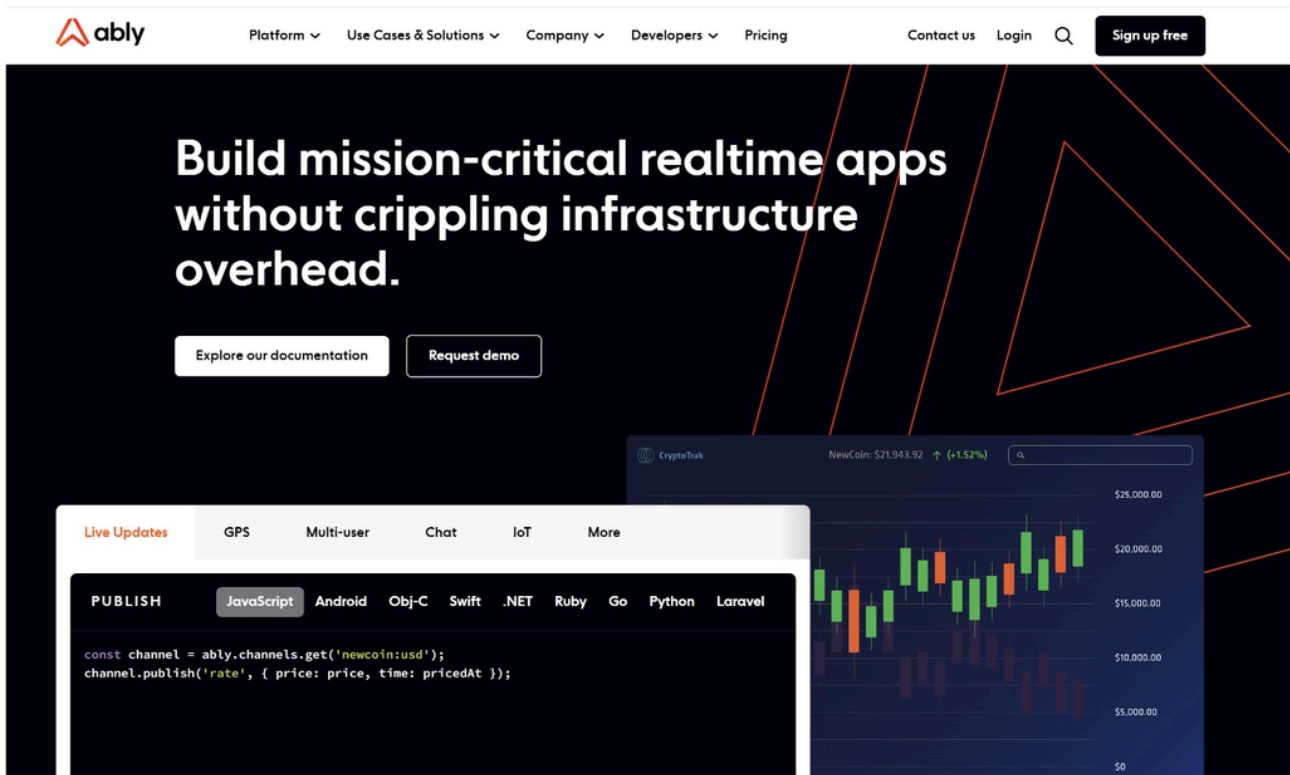
명령제어를 위해 Aply라는 Third party 라이브러리를 악용해 명령제어를 위한 비밀 통신 채널을 생성하였으며, 해당 악성 프로그램을 Go언어로 작성해 유포했습니다. 뿐만 아니라 목표대상에 침투하고 지속성을 유지(채널이 동작하지 않을 상황을 대비)하기 위해 다양한 명령제어를 위한 악성코드를 추가로 설치하고 있음을 확인했습니다.

우리는 이번 보고서를 통해 ScarCruft 그룹의 신규 악성코드인 Go언어 기반 원격제어기를 이용한 명령제어 과정과 침투 이후 공격자의 행위(Procedures)에 대해 상세히 이야기합니다.

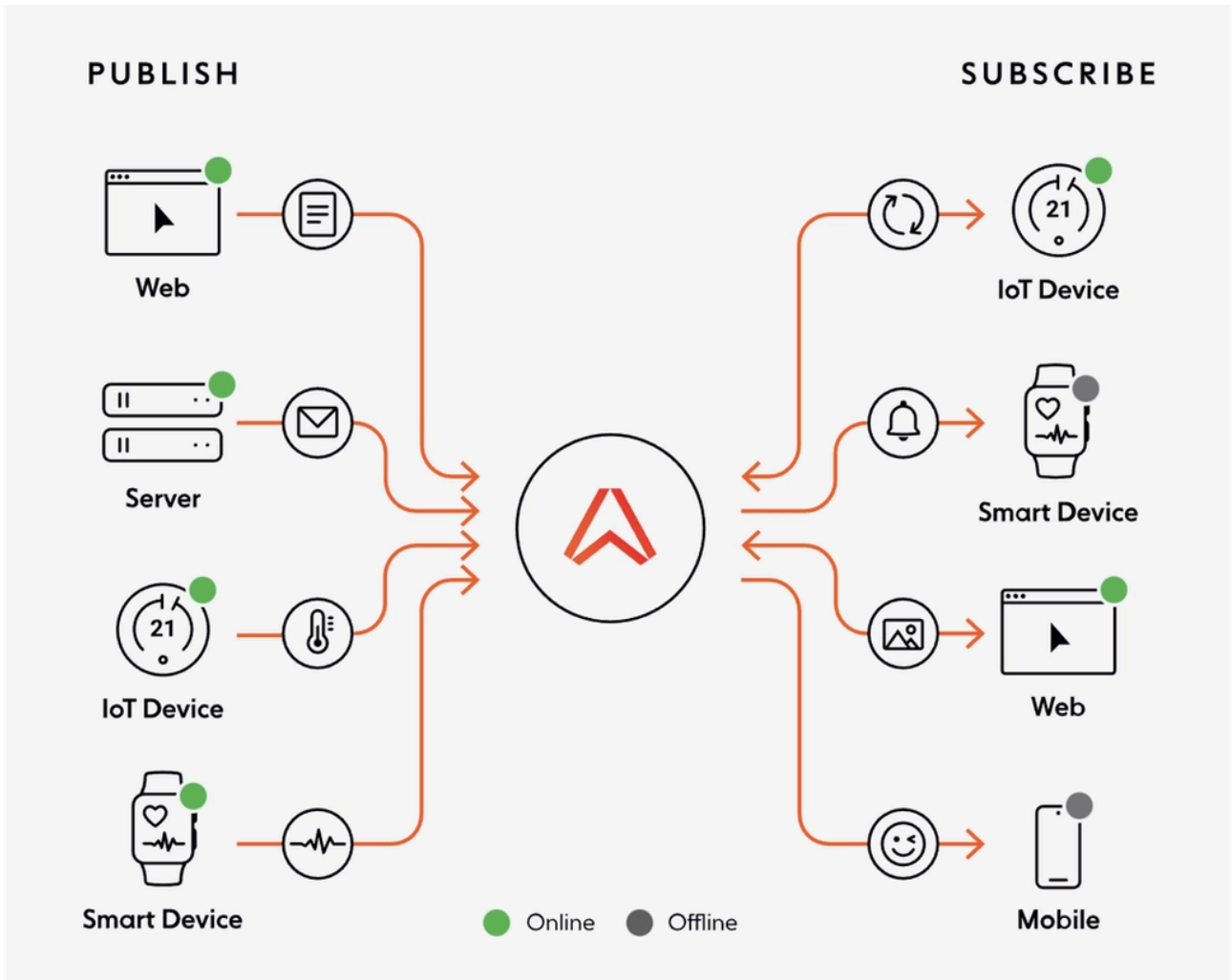




Ably는 실시간으로 데이터를 송수신하여, 통계정보를 제공하거나 휴대폰 알림 서비스 등으로 활용할 수 있는 서비스입니다. Ably 서비스를 이용하면, 서버의 메일과 스마트워치의 생체정보도 저장하고 이를 실시간으로 연동하여 데이터를 공유할 수 있기 때문에 유연한 정보 송수신이 가능합니다. 공격자는 Ably 서비스를 악용하여, Go 언어 악성코드에 명령제어 기능을 새로이 구현했습니다.

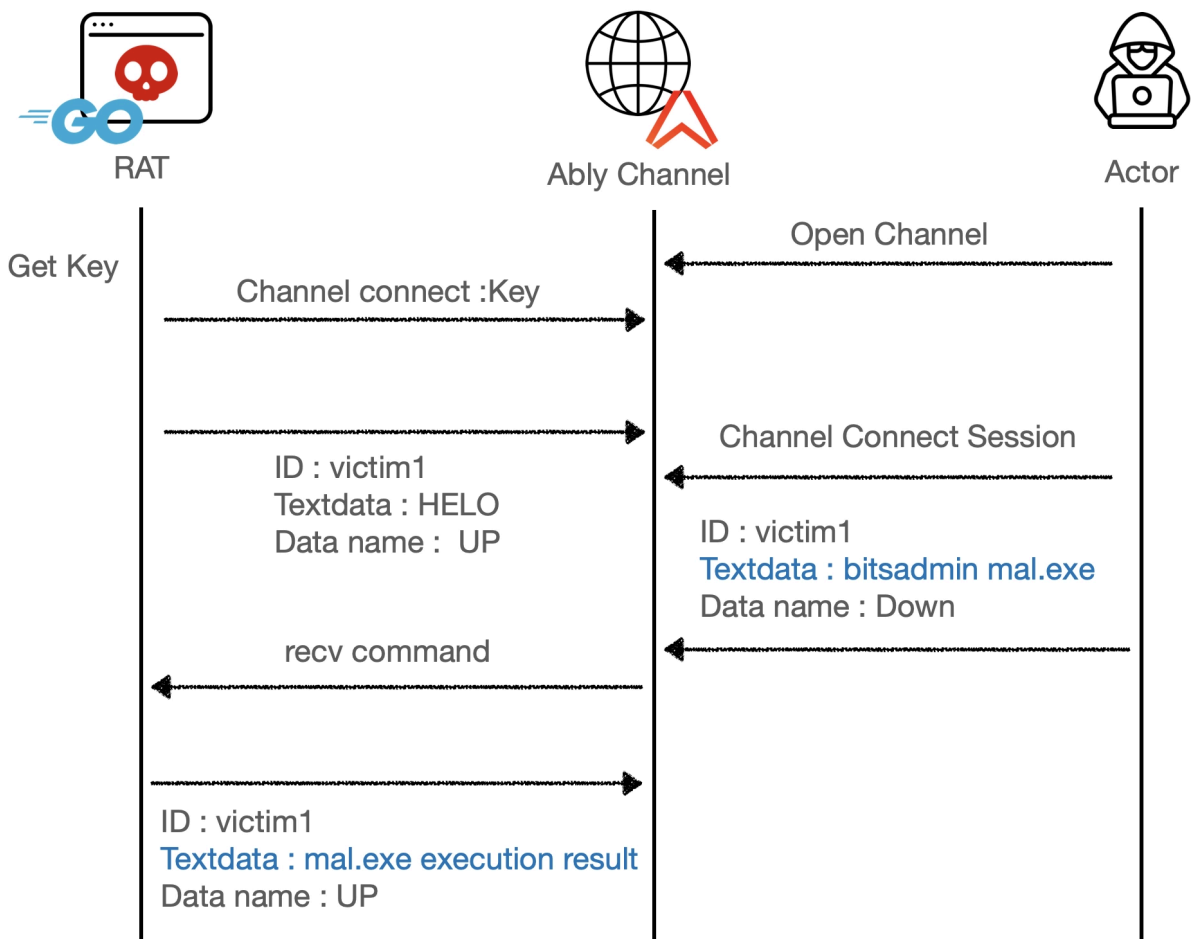


Ably(<https://ably.com>)[2]



### Ably 서비스 개요도

공격자는 Ably 서비스를 악용하여 새로운 명령제어 체계를 구축하였습니다. 그리고 이 체계는 Golang 언어를 이용하여 제작된 새로운 악성코드를 기반으로 동작합니다. 간단한 동작절차는 다음과 같습니다.



공격자는 Ably 채널을 생성하고 감염자의 채널접속을 대기합니다.

피해자는 악성코드에 감염되면 내부에 존재하는 Ably API 키와 함께, PC 이름과 계정명으로 HELO라는 문자를 주기적으로 채널에 업로드(데이터명: UP)합니다.

모니터링하고 있던 공격자는 명령이 필요한 PC에게 추가 악성코드를 감염시키는 명령을 base64로 인코딩하여 하달(데이터명: DOWN)합니다.

피해자는 채널을 실시간으로 접근하여 자신의 ID가 있는 명령을 확인, 실행합니다.

피해자는 실행결과를 다시 채널로 업로드(데이터명: UP)합니다.

이 명령체계의 장점 중 첫번째는 차단이 불가능 하다는 점입니다. 우리는 정상적인 서비스인 Ably를 차단할 수 없기 때문입니다. 두번째는 공격자의 IP 노출이 최소화된다는 점입니다. Ably 내부에 로그가 존재할 수는 있지만, 서비스 이용자에게 접속 로그를 제공하는 기능은 존재하지 않습니다. 그리고 세번째는 외부 서비스를 사용하기 때문에 어디에서나 명령제어가 가능하다는 점입니다. 위와 같이, 공격자는 Ably 서비스의 장점을 그대로 활용할 수 있습니다.

중요한 명령을 하달할때는 ScarCruft의 대표적인 원격제어도구인 Chinotto 악성코드 뿐 아니라 다른 환경에서 동작가능한 원격제어도구를 개발/운용하는 것으로 추정됩니다.

```
.text:00000000077C71C      nop     dword ptr [rax+00h]
.text:00000000077C720      call   github_com_ably_ably_go_ably_WithKey
.text:00000000077C725      mov     rax, [rsp+60h+var_50]
.text:00000000077C72A      mov     [rsp+60h+var_10], 0
.text:00000000077C733      mov     [rsp+60h+var_10], rax
.text:00000000077C738      lea    rax, [rsp+60h+var_10]
.text:00000000077C73D      mov     [rsp+60h+var_60], rax
.text:00000000077C741      mov     [rsp+60h+var_58], 1
.text:00000000077C74A      mov     [rsp+60h+var_50], 1
.text:00000000077C753      call   github_com_ably_ably_go_ably_NewRealtime
.text:00000000077C758      mov     rax, qword ptr [rsp+60h+var_48]
.text:00000000077C75D      mov     rax, [rax+8]
.text:00000000077C761      mov     [rsp+60h+var_60], rax
.text:00000000077C765      lea    rax, unk_835DDC
.text:00000000077C76C      mov     [rsp+60h+var_58], rax
.text:00000000077C771      mov     [rsp+60h+var_50], 6
.text:00000000077C77A      xorps  xmm0, xmm0
.text:00000000077C77D      movups [rsp+60h+var_48], xmm0
.text:00000000077C782      mov     [rsp+60h+var_38], 0
.text:00000000077C78B      call   github_com_ably_ably_ably__RealtimeChannels_Get
.text:00000000077C790      mov     rax, [rsp+60h+var_30]
.text:00000000077C795      mov     [rsp+60h+var_18], rax
.text:00000000077C79A      lea    rcx, unk_7F4B80
```

악성코드 내부에서 활용되는 Ably API

주소	Hex	ASCII
0000000008416E5	48 41 45 58 77 67 2E 30 35 63 33 30 51 3A 73 31	HAEXwg.05c30Q:s1
0000000008416F5	74 67 71 50 70 72 47 6A 65 75 4F 64 47 4E 48 45	tgqPprGjeuOdGNHE
000000000841705	41 44 45 52 53 20 66 72 61 6D 65 20 77 69 74 68	ADERS frame with
000000000841715	20 73 74 72 65 61 6D 20 49 44 20 30 4D 61 70 49	stream ID OMapI
000000000841725	74 65 72 2E 4B 65 79 20 63 61 6C 6C 65 64 20 62	ter.Key called b
000000000841735	65 66 6F 72 65 20 4E 65 78 74 50 61 63 69 66 69	efore NextPacifi

Realtime 클라이언트 객체 생성을 위한 Ably API 키

주소	Hex	ASCII
000000000835DDC	67 6C 6F 62 61 6C 67 6F 70 68 65 72 68 61 6E 67	globalgopherhang
000000000835DEC	75 70 68 65 61 64 65 72 68 65 69 67 68 74 68 69	upheaderheighti
000000000835DFC	64 64 65 6E 69 6D 70 6F 72 74 69 70 2B 6E 65 74	ddenimportip+net
000000000835E0C	6A 73 43 74 78 28 6B 69 6C 6C 65 64 6C 69 73 74	jsCtx(killedlist
000000000835E1C	65 6E 6D 61 6D 6C 74 6E 6D 65 74 68 6E 64 6D 6D	emailtomethodmi

공격자가 명령제어로 악용할 Ably 채널 이름(global)

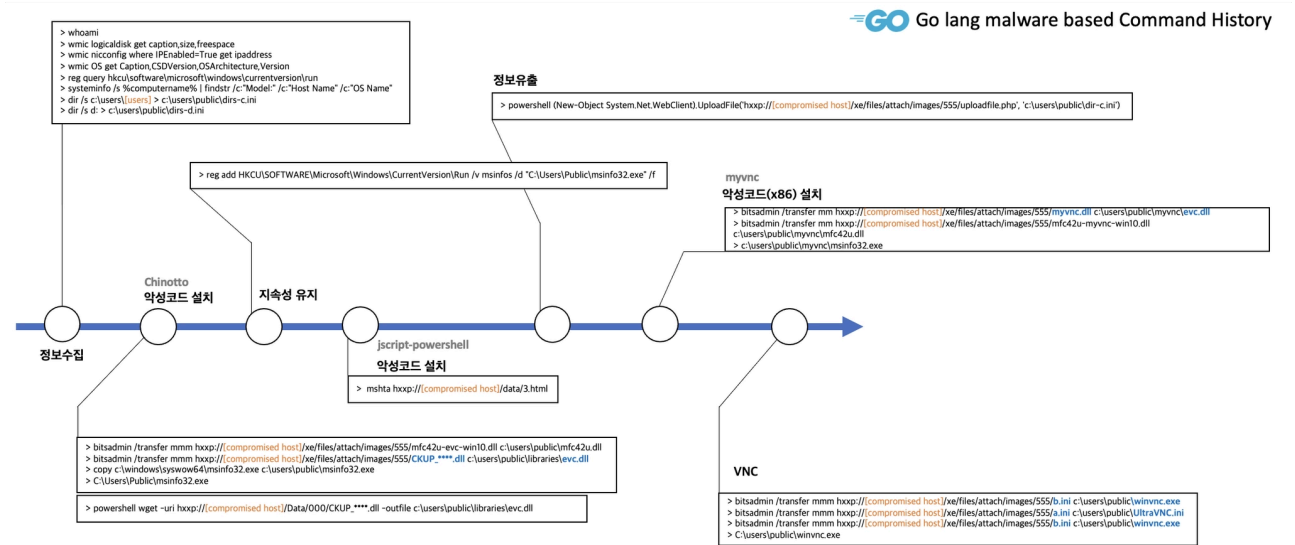
주소	Hex	ASCII
000000000835493	44 4F 57 4E 44 61 73 68 44 61 74 65 45 45 53 54	DOWNDashDateEST
0000000008354A3	45 74 61 67 46 72 6F 6D 47 45 54 20 47 4F 47 43	EtagFromGET GOGC
0000000008354B3	47 6F 6E 65 48 6F 73 74 49 4D 41 47 49 4E 46 4F	GoneHostIMAGINFO
0000000008354C3	4A 75 6C 79 4A 75 6E 65 4C 45 41 46 4C 69 73 75	JulyJuneLEAFLisu
0000000008354D3	4D 69 61 6F 4D 6F 64 69 4E 5A 44 54 4E 5A 53 54	MiaoModiNZDTNZST
0000000008354E3	4E 65 77 61 50 49 4E 47 50 4F 53 54 53 41 53 54	NewaPINGPOSTSAST
0000000008354F3	53 74 61 74 54 68 61 6D 55 6D 6E 74 57 41 52 4E	StatThaiIntWAPN

클라이언트가 공격자 명령을 수신하는 데이터 이름(DOWN)

우리는 선제적으로 대응을 하는 Defend Forward 관점에서 3.2에서 확인할 수 있었던 Ably 서비스의 API 키를 활용, 공격자의 명령제어 채널을 모니터링할 수 있었습니다. 특히, 악성코드를 단순 실행하게 되면 분석환경 정보가 공격자에게 송신되기 때문에, Go 언어를 기반으로 모니터링 스크립트를 직접 개발, 운영하였습니다. 그 결과 ScarCruft의 명령제어 데이터를 확보, 추가적으로 데이터를 분석할 수 있었습니다.

Defend Forward는 방어자가 공격적인 사고방식으로 사이버 공격에 대응하는 것을 의미하며, 공격 초기단계에서 선제적 대응을 통해 공격자 리소스를 증가시키는 등 방어활동을 수행합니다.

ScarCruft의 명령제어를 모니터링하는 중 다양한 유형의 명령제어 악성코드를 확인 가능했으며, 과거부터 활용되어 온 명령제어 악성코드와 Golang으로 제작된 신규 서비스(Ably) 기반 악성코드까지 활용한 것을 발견했습니다. 이것은 메인 명령제어 체계의 수단이 비정상적으로 동작할 경우 대체하기 위한 면도 있겠지만, 잠재적 공격대상들이 보유한 환경이 다양하기 때문에 그에 따라 공격도구를 다변화하여 사용하는 것으로 추정됩니다.



File Name	Description	comment
CKUP_****.dll	정보유출 및 명령제어(스크린 캡처, 키로깅 등)	PE
Second.ini	PowerShell 기반 명령제어기	GO BASE PE
3.html	Second.ini 와 동일 기능	JScript
myvnc.dll	Second.ini 와 동일 기능	PE
a.ini	Ultra VNC 관련 설정 파일	Setting file
b.ini	UltraVNC Server Property Page	PE

수집 로그('22.10.18 16:57:35~'22.12.21 02:16:35, 78,834라인, 5.9MB, 18개 감염기기)

최초 감염 이후, 공격자는 어떠한 명령을 수행하는가?

감염단말기로 개인별로 특화된 악성코드를 내리고, msinfo32.exe를 통해 악성코드를 실행, 공격자와 감염자의 통신채널을 구축합니다. 이후, 정보유출을 위해서 UltraVNC를 사용하기도 합니다.

아래는 감염자가 최초로 감염된 이후에 공격자가 어떻게 명령을 내렸는지 확인할 수 있는 실제 데이터입니다.

> Loading Plain Text code...

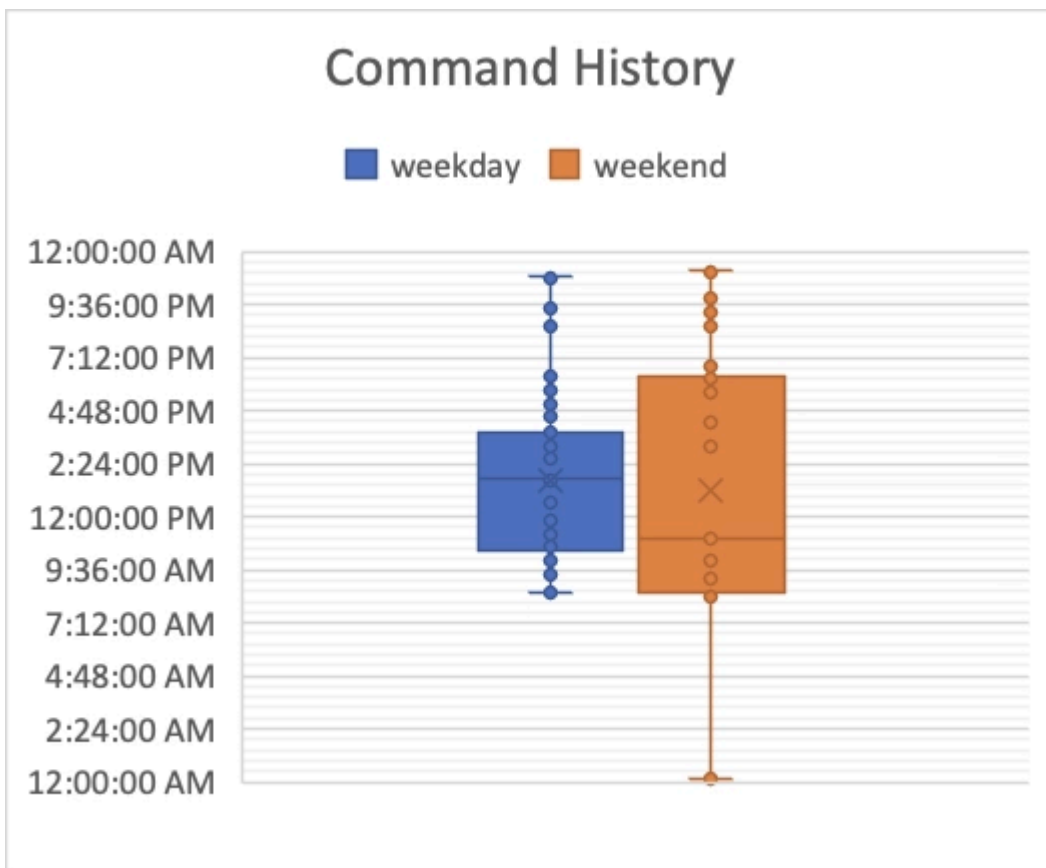
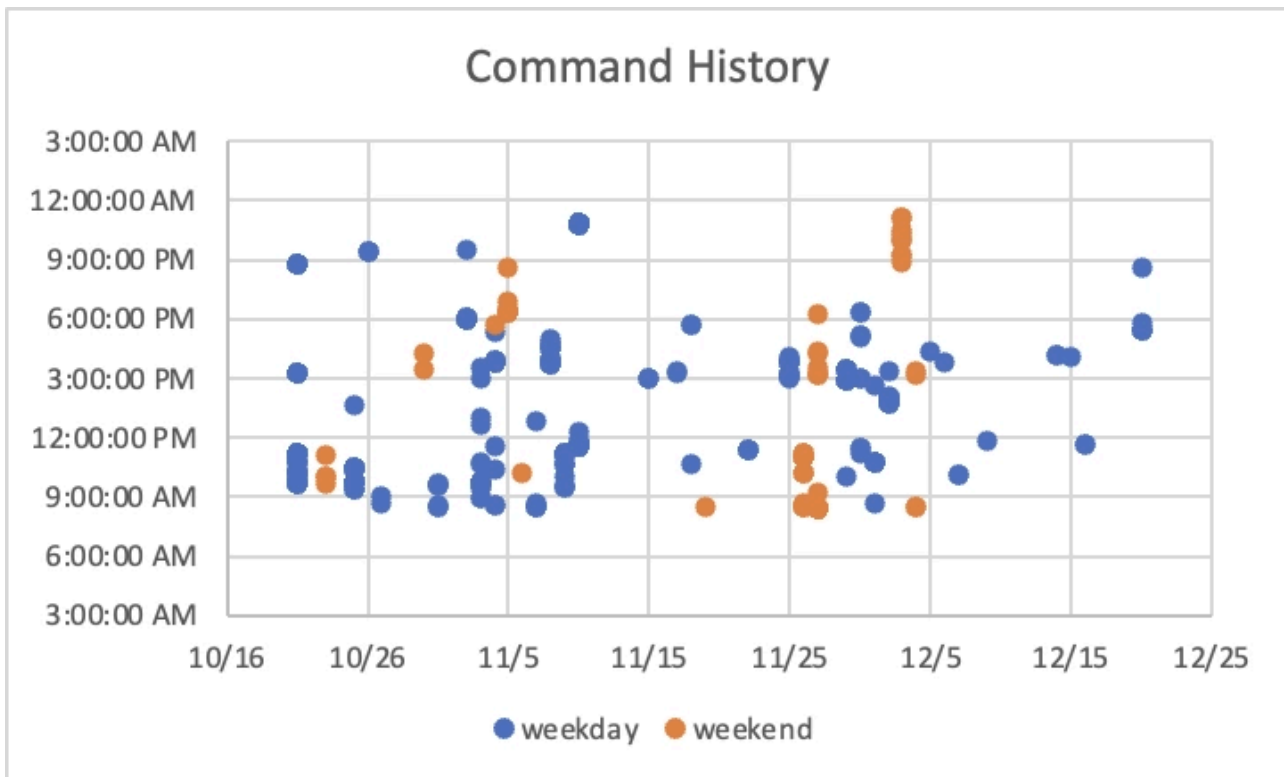
> Loading Plain Text code...

공격자는 언제 작업을 수행하는가?

공격자는 평일에 근무하고 주말에 휴식을 취하는 주6일제로 근무하는 것으로 보입니다. 또한, KST(UTC+9)를 기준으로 데이터를 해석할 경우, 업무시간 또한 일반적인 9 TO 6로 추정됩니다(혹은 더 이른 출근시간으로 보입니다).

모니터링 했던 2개월여 기간 중 명령이 없던 날은 9일(전체 주말: 18일)이기 때문에 평균적으로 주 6일로 근무한다고 추정할 수 있습니다. 또한, 휴일을 탄력적으로 조정가능한 것으로 보입니다.

감염기기(DESKTOP-\*\*\*\*\*-user)의 경우, 공격자는 토요일 저녁(2022-12-03 20:51:38)에 처음으로 작업을 시작하여, 자정(2022-12-04 00:06:29)에 공격을 마무리하고, 다음날 일요일인 2022-12-04 08:31:38에 다시 공격을 시작하였습니다. 위의 사례를 보아 특별근무가 필요한 경우, 공격자는 주말에도 쉬지않고 공격하는 것으로 보입니다.



공격자는 최초 침투 이후 어떤 작업을 수행하는가?

공격자는 윈도우 실행시 mshta를 통해 악성코드가 실행되도록 레지스트리에 등록합니다. 이 스크립트는 자동화된 프로그램에 의해 생성되었을 수 있다고 추정합니다.

ping 명령어의 수행시간과 레지스트리 등록 값을 랜덤으로 부여하며, 레지스트리에 등록되는 랜덤 값들이 동일한 값을 가지는 것은, mshta를 통해 실행되는 악성 스크립트가 프로그램에 의해 새로 생성된다고 추정할 수 있습니다.

> Loading Plain Text code...

아래 근거를 종합하여 판단하였을때, ScarCruft의 직원들에게는 공격을 수행하기 위한 매뉴얼과 스크립트 생성기가 존재하고, 일부 변수에 대하여 사람이 직접 수정, 공격하는 것으로 추정됩니다.

TTPs #9 개인의 일상을 감시하는 공격전략 분석

Ably

---

Source: <https://thorcert.notion.site/TTPs-ScarCruft-Tracking-Note-67acee42e4ba47398183db9fc7792aff>