

Trigona Rebranding Suspicions and Global Threats, and BlackNevas Ransomware Analysis

9/9/2025



BlackNevas has been continuously launching ransomware attacks against companies in various industries and countries, including South Korea. This post provides a technical analysis on the characteristics, encryption methods, and reasons why BlackNevas encrypts files in a way that makes them impossible to decrypt. It is hoped that this post will provide insights for defending against similar threats in the future.

1. Overview

1.1. BlackNevas

The BlackNevas ransomware group first appeared in November 2024 and has since been continuously attacking various businesses and critical infrastructure organizations in Asia, North America, and Europe. Like other ransomware groups, BlackNevas encrypts files on infected systems, steals sensitive data from

affected companies, and threatens to leak the data if the ransom is not paid. The group’s targets are spread across multiple regions, with a particular focus on the Asia-Pacific region (50%). Major targets include countries in Southeast Asia and East Asia such as Japan, Thailand, and Korea. In Europe, the group targets countries in Western Europe and the Baltic Sea region, including the UK, Italy, and Lithuania. In North America, the group has targeted Connecticut in the United States.

The BlackNevas ransomware does not officially operate under the Ransomware-as-a-Service (RaaS) model. They threaten victims by claiming that the breach of their data will be handed over to their own data leak site (DLS) and affiliated partners. The threat actors encrypt files using a combination of AES (symmetric key) and RSA (public key). The extension “.-encrypted” is added to the encrypted files.

The ransomware distributed by the BlackNevas group does not perform any additional network communication after encrypting files. The ransom note instructs victims to contact them via email or Telegram to decrypt the encrypted data.

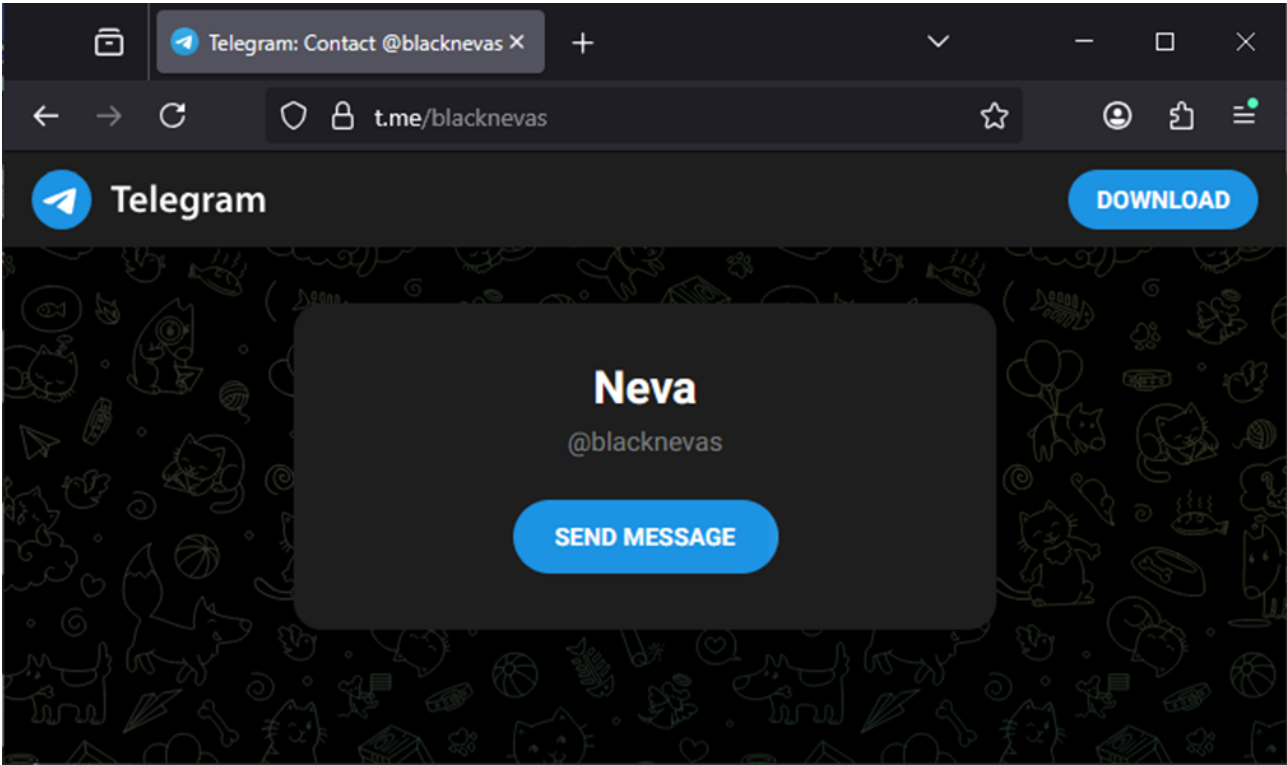


Figure 1. Threat actor’s Telegram address within the ransom note

2. Analysis

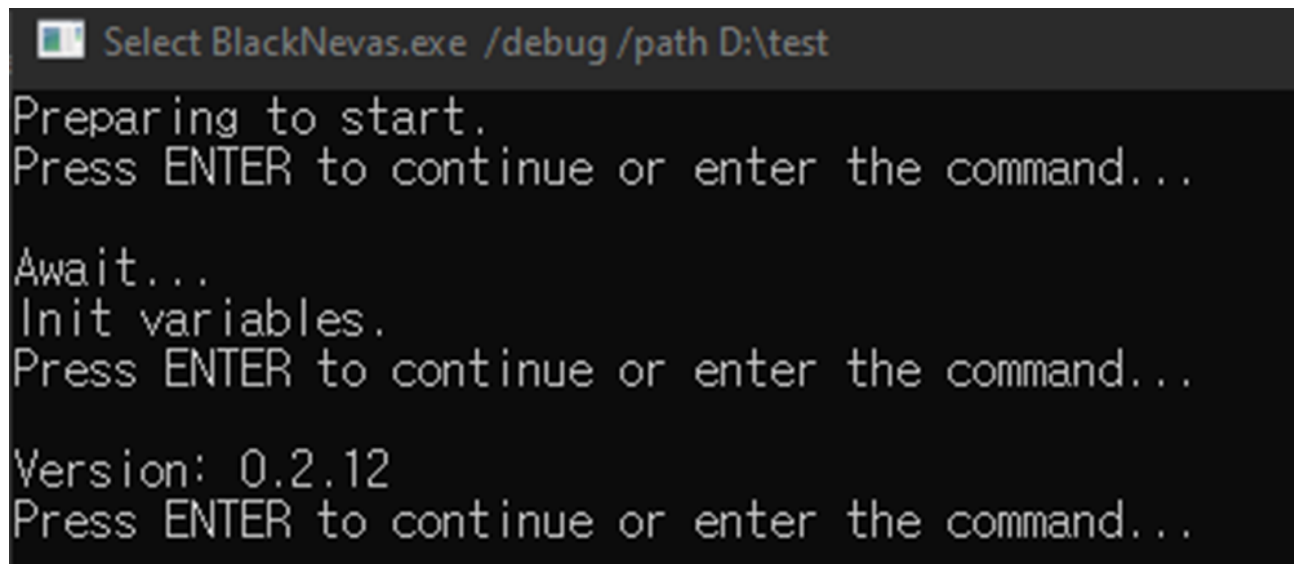
2.1. Initial Routine

The BlackNevas ransomware did not include anti-debugging techniques or anti-sandbox evasion features. Instead, it supports multiple arguments, and the details of these arguments are shown in Table 1 below.

Argument	Behavior
/allow_system	Encrypts system path
/fast	Encrypts only 1% of the entire file size
/full	Encrypts entire file
/path	Specifies the encrypted path
/debug	Outputs execution logs when encrypted
/stealth	Changes extension upon encryption and creates ransom note
/shdwn	Terminates system after file encryption

Table 1. Behavior according to the parameter value

As shown in Table 1, the BlackNevas ransomware outputs its execution history via the “/debug” argument, which also reveals the ransomware’s version information. Additionally, when neither the “/fast” nor “/full” argument is specified, only the first 10% of the file’s full size is encrypted from the beginning.



```
Select BlackNevas.exe /debug /path D:\test
Preparing to start.
Press ENTER to continue or enter the command...

Await...
Init variables.
Press ENTER to continue or enter the command...

Version: 0.2.12
Press ENTER to continue or enter the command...
```

Figure 2. Version information output through the “/debug” argument

2.2. Preparation for Encryption

BlackNevas ransomware does not encrypt paths that are inaccessible during the encryption process or paths that contain the string “system32” or “windows”. All other accessible paths are subject to encryption. Typically, other ransomware operate based on a predefined list of paths to exclude from encryption. In contrast, BlackNevas ransomware features a characteristic in which it conditionally checks path strings at runtime to determine whether they should be excluded from encryption.

```

System::_WStrCat3(vars28, v14, L"system32");
if ( !sub_543080(vars28[0]) )
{
    v1 = 0;
    if ( v14 )
        v1 = *(v14 - 1) >> 1;
    v2 = *(off_597F90 + 29);
    v3 = 0;
    if ( v2 )
        v3 = *(v2 - 4) >> 1;
    if ( v1 < v3 )
        goto LABEL_14;
    v4 = 1;
    v5 = 0;
    if ( v2 )
        v5 = *(v2 - 4) >> 1;
    v6 = v5;
    v7 = 1;
    if ( v5 >= 1 )
    {
        while ( v14[v7 - 1] == (*(off_597F90 + 29) + 2LL * v7 - 2) )
        {
            ++v7;
            if ( !--v6 )
                goto LABEL_13;
        }
        v4 = 0;
    }
LABEL_13:
    if ( !v4 )
    {
LABEL_14:
        v8 = sub_412E50(&dword_54262C, v14, 1);
        if ( v8 )
        {
            while ( 1 )
            {
                System::_WStrCopy(vars38, v14, 1, (v8 - 1));
                if ( !sub_412550() )
                    // Path Check "Windows"

```

Figure 3. Code to check the exclusion path of encryption

To prevent the system from being damaged by encrypting key files, specific extensions and files are excluded from encryption. The files excluded from encryption are “NTUSER.DAT” and the ransom note “how_to_decrypt.txt”, while the extensions excluded from encryption are shown in Table 2 below.

File extensions excluded from encryption

sys, dll, exe, log, bmp, vmem, vswp, vmxf, vmsd, scoreboard, nvram, vmss

Table 2. File extensions excluded from encryption

2.3. File Encryption

There are two types of file name changes when encrypting files: “random name.random name.-encrypted” and “trial-recovery.random name.random name.-encrypted”. The Table 3 below shows the extensions that include “trial-recovery” in the file name. In other cases, the file name is changed to the “random name.random name.-encrypted” format. As can be inferred from the file name, the type with the “trial-recovery” prefix is likely given to certain files as a demonstration to the victims to show that the files can actually be decrypted.

Extensions that use “trial-recovery” in the file name

doc, docx, hwp, jpg, pdf, png, rtf, txt

Table 3. Extensions that use “trial-recovery” in the filename

Once the check of the exclusion path, file, and extension is completed, it checks whether the file is already encrypted before encrypting it. Typically, most ransomware use the method of designating the extension of files that have already been encrypted as an exception extension to determine the infection status. However, BlackNevas ransomware does not use this method. Instead, it compares specific data values to determine whether a file has been encrypted.

As shown in Figure 4 below, the 8-byte value at the end of the original file before encryption is checked (highlighted in red) to determine if it is the “E” or “R” type. “E” refers to the type where the file extension is changed after encryption, and “R” refers to the type where the file name is changed to “trial-recovery”. The same 8-byte value at the end of the encrypted file is then checked, which indicates the size of the additional data added to the original file size.

Upon completion of the infection check, the file is encrypted using an AES symmetric key as mentioned in the overview. The generated AES key is then encrypted with an RSA public key and inserted at the end of the file. As a result, there are no clues left behind in the local environment that can be used to decrypt the file.

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text	
00000000	00	45	00	45	07	45	88	45	00	02	06	00	00	00	16	2D	.E.E.E^E.....-	Encryption Check (HEAD)
00000010	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	2D	-----	
00000020	2D	2D	2D	2D	2D	00	00	00	07								-----....	Victim ID
00000030	00	00	00	07	52	4F	42	4F	43	4F	50	00	00	02	87	00	ROBOCOP...+	
00000040	00	02	02	2C	C8	5D	82	28	64	04	48	38	E5	4F	84	E2	...,E],(d.H8dO.,ã	
00000050	5D	30	FB	1D	2C	6D	D8	0D	CA	DF	C2	8E	12	96	BF	E1	]0û.,m0.ÊBÂŽ.-zÁ	Encrypted (AES KEY + IV) using RSA Public Key
00000060	4D	D7	9F	56	26	B1	DF	1A	09	4E	03	8F	D3	B6	C8	47	M×YV&±B..N..ÓqEG	
00000070	61	3B	7A	63	6E	F7	C8	01	ED	72	66	8C	7F	49	55	F5	a;zcñ÷Ê.irfE.IUð	
00000080	E7	8F	C1	25	03	AF	D1	83	3A	0C	8D	B2	B2	A5	AF	75	ç.Â%.~Ñf:..^"u	
00000090	0A	78	4E	53	19	33	53	77	99	92	A4	BF	B6	E6	77	79	.xNS.3Sw"r;çqæwy	File size before encryption
000000A0	DC	44	97	6F	85	33	E2	C1	78	C4	98	49	46	11	33	DD	ÜD-o...3âÄxÄ~IF.3Y	
000000B0	C8	21	1D	79	92	4F	37	6B	13	98	62	C8	25	93	AF	86	È!.y'O7k."bE%"~t	Encryption Check(Header)
000000C0	A2	86	A8	A0	1C	0A	4E	A8	F7	9D	12	20	F9	82	E3	03	çt" ..N"÷... ù,â.	
000000D0	84	7A	61	0A	9D	95	0A	76	EE	22	6C	EC	FD	5E	8D	72	„za...vi"liý^r	
000000E0	B7	2E	C8	43	06	67	E9	9A	51	3A	F1	F4	25	A1	E7	A6	..ÈC.gésQ:ñô%;ç!	
000000F0	36	30	F8	2C	D9	86	8B	95	7C	20	9C	AF	58	A2	45	53	60æ,Üt<• æ`XcES	
00000100	14	11	10	8A	11	46	E4	0F	BD	81	40	FD	CB	47	AD	F8	...Š.Fä.%.@ýEG.ø	
00000110	DD	49	0E	03	FD	3E	54	A1	F4	8E	17	94	54	8A	21	5D	ÝI..ý>T;ôŽ."TŠ!]	
00000120	27	2B	74	79	52	BB	0F	D0	E6	C5	08	7B	BE	33	93	88	'+tyR».DæÄ.(%3"~	
...																		
000006D0	2D	31	00	78	C9	8E	29	03	44	5D	A4	A8	E8	36	3D	6C	-1.xEŽ).D]H"è6=1	
000006E0	18	0E	1C	56	64	36	71	8F	C1	83	BC	9C	1A	E3	E0	E3	...Vd6q.Äf*æ.äää	
000006F0	47	81	63	74	59	D3	37	90	F2	DA	92	B7	83	6E	03	FD	G.ctYÓ7.òÚ'·fn.ý	
00000700	12	5B	85	92	57	02	9A	02	1A	1D	D4	58	E0	73	D1	3C	.[...'W.š...ÔXàsN<	
00000710	A5	45	27	10	FD	69	6A	71	A7	A2	12	12	26	7A	CE	E9	¥E'.ýijq\$ç...æzIé	
00000720	4E	D0	41	65	9B	92	DF	9C	1E	E7	36	46	84	D2	22	BA	NDAe>'Bæ.ç6F„Ö"o	
00000730	73	D7	4C	21	4A	00	49	67	E9	FE	E5	06	2D	16	7D	AE	s×L!J.Igépâ.-.)@	
00000740	EC	74	7E	94	E2	39	70	17	02	B5	26	BF	4E	4C	00	00	it~"â9p..µ&¿NL..	
00000750	00	1F	F0	02	86	47	54	A6	32	FC	B3	2B	84	3E	DD	20	..ð.†GT;2ú'+„>Ý	
00000760	B2	73	C8	F1	18	A5	C2	F2	98	D5	1E	D6	EF	B6	3F	A2	°sÈñ.¥Äò~Ô.Ôiŕ?c	
00000770	9C	00	00	00	00	10	00	00	00	00	00	00	00	00	00	00	æ.....*°	
00000780	00	00	07	88	00	00	07	88									^..^	

Figure 4. Data structure added after file encryption

2.4. Ransom Note

The following is a screen captured after being infected by BlackNevras ransomware. The desktop background has not been changed.



Figure 5. Test environment after encryption is complete and the desktop is changed

The ransom note is created under the name “how_to_decrypt.txt”. As the ransom note is created in all folders except for the exclusion folder, the ransom note is created in all folders present in the system. The note claims that the threat actor is a professional in file encryption and industrial espionage activities. It also includes a blackmail message threatening to leak the victim’s data to their partners, post it on a blog, or put it up for auction if a decryption negotiation is not made within 7 days.

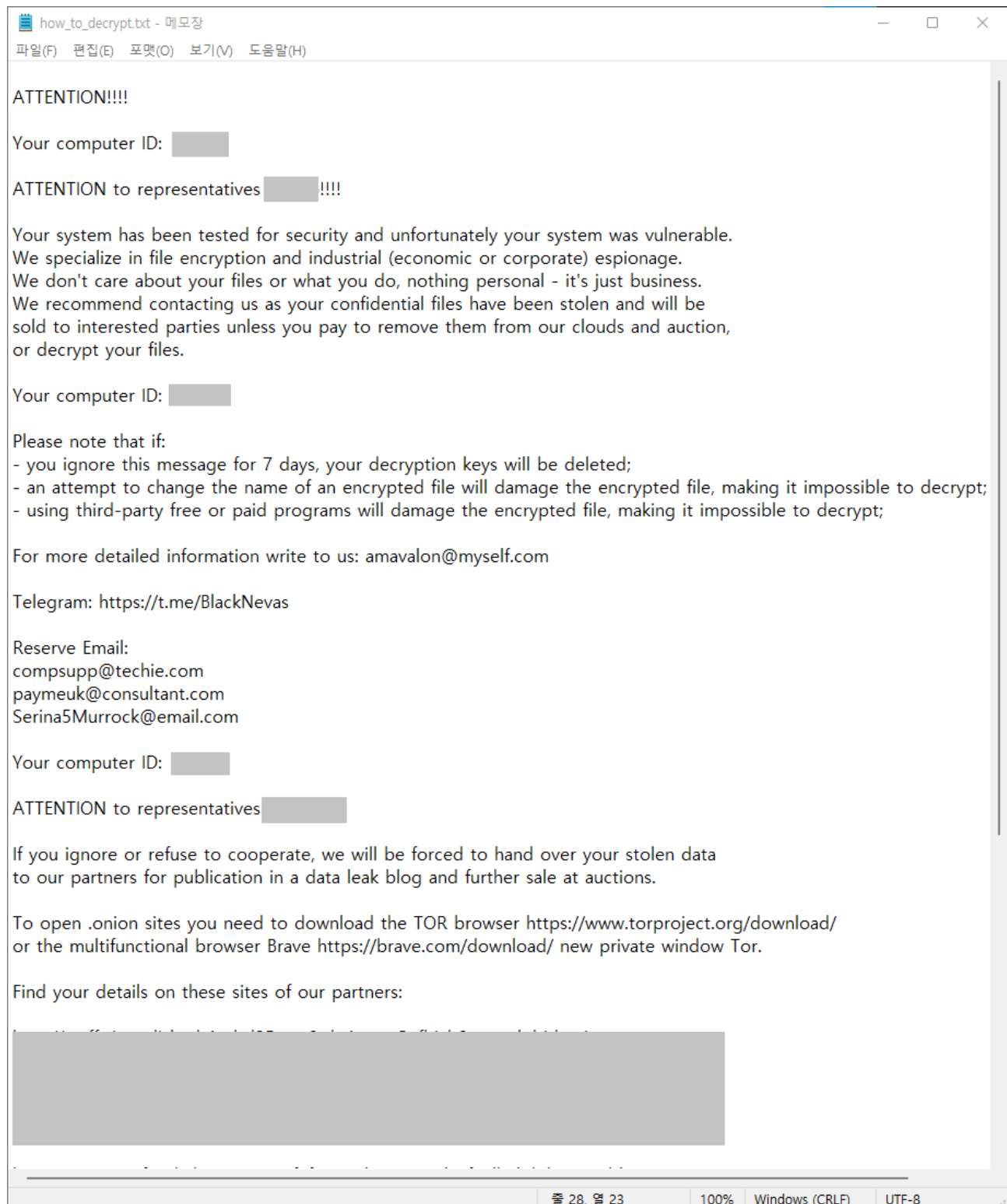


Figure 6. Ransom note (how_to_decrypt.txt)

3. Conclusion

The number of companies suffering ransomware attacks through the DLS of the BlackNevan group is on the rise. This group is distributed globally and has been particularly targeting the Asia-Pacific region (50%). The primary targets are countries in Southeast and East Asia, such as Japan, Thailand, and Korea. In Europe, the group has been targeting countries along the Baltic Sea, such as the UK, Italy, and Lithuania. In North America, the group has been targeting Connecticut in the US. As described in this report, the ransomware encrypts files using an AES symmetric key, and then encrypts the AES key with an RSA public key. Therefore, unless the RSA algorithm itself is broken, there is no possibility of decryption. For this reason, companies are strongly advised to strictly adhere to the following response guide to protect and stably operate their key assets.

4. AhnLab's Response

The diagnostic names and engine dates of the AhnLab product groups are as follows:

4.1. V3

Ransomware/Win.Trigona.R585545 (2025.08.09.00)

Ransom/MDP.Decoy.M1171 (2016.07.15.02)

4.2. EDR

Ransom/EDR.Decoy.M2470 (2022.09.30.00)

Ransom/EDR.Event.M11760 (2024.06.19.02)

MD5

2374998cffb71f3714da2075461a884b

4a1864a95643b0211fa7ad81b676fe2e

9f877949b8cbbb3adfe07fd4411b9f26

f2547a80dd64dcd5cba164fe4558c2b6

Additional IOCs are available on AhnLab TIP.



AhnLab TIP

Stay Ahead of Rapidly Evolving Threats Make the Best-Informed Decisions

Get Started with AhnLab's State-of-the-Art Threat Intelligence

atip.ahnlab.com