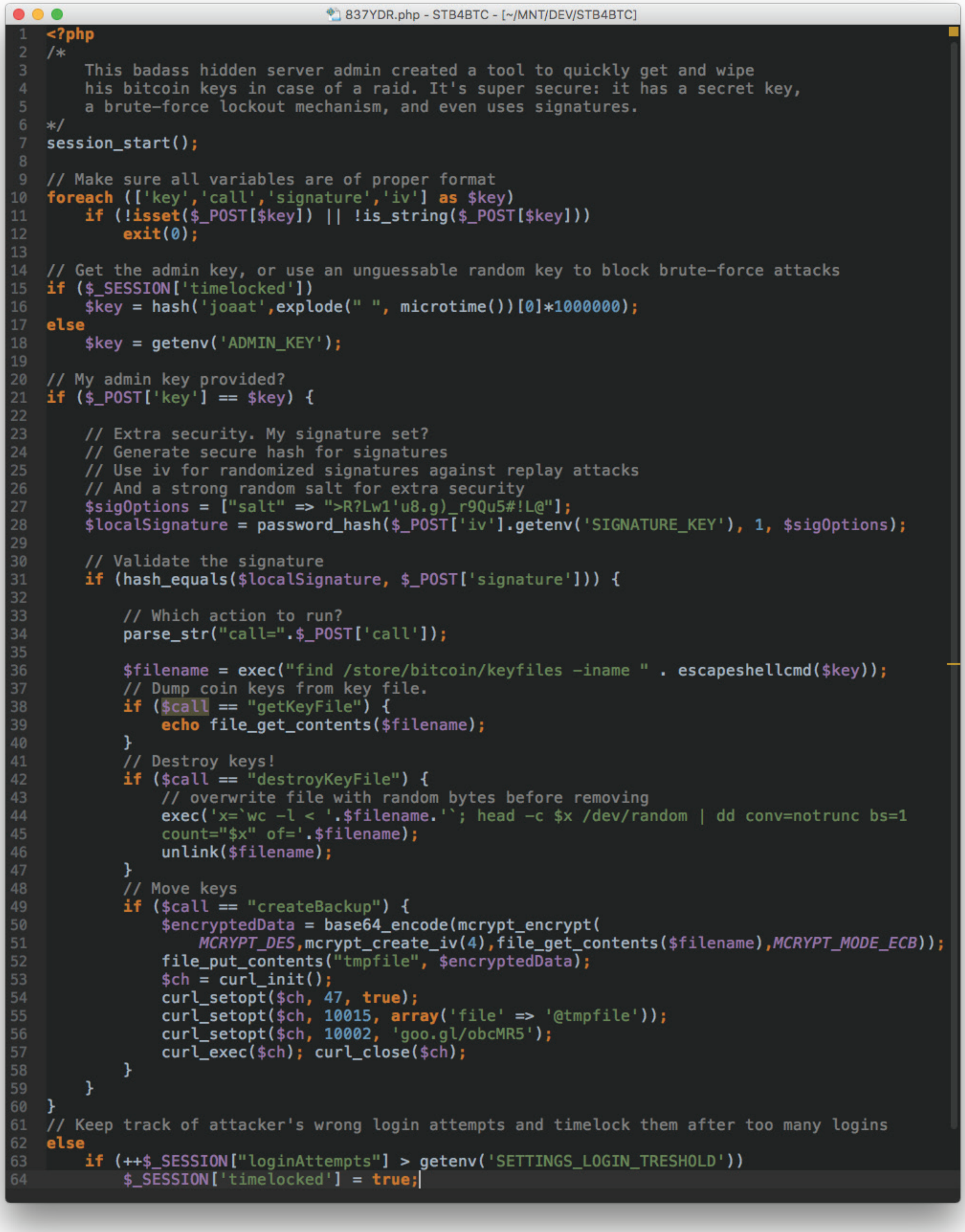
Seconds before his Tor server got seized, the admin managed to wipe his bitcoin keys and a bunch of other evidence. Altough he got his business up and running again on a new hidden server, the admin freaked out about the bust. He asked his friends to code-review his admin script to identify any possible security defects! The one that spots the most flaws/bugs is awarded a bitcoin!

Mail your submission or questions to stb@securify.nl. Challenge closes 01-01-2017.



```php
<?php
/*
    This badass hidden server admin created a tool to quickly get and wipe
    his bitcoin keys in case of a raid. It's super secure: it has a secret key,
    a brute-force lockout mechanism, and even uses signatures.
*/
session_start();

// Make sure all variables are of proper format
foreach (['key','call','signature','iv'] as $key)
    if (!isset($_POST[$key]) || !is_string($_POST[$key]))
        exit(0);

// Get the admin key, or use an unguessable random key to block brute-force attacks
if ($_SESSION['timelocked'])
    $key = hash('joaat',explode(" ", microtime())[0]*1000000);
else
    $key = getenv('ADMIN_KEY');

// My admin key provided?
if ($_POST['key'] == $key) {

    // Extra security. My signature set?
    // Generate secure hash for signatures
    // Use iv for randomized signatures against replay attacks
    // And a strong random salt for extra security
    $sigOptions = ["salt" => ">R?Lw1'u8.g)_r9Qu5#!L@"];
    $localSignature = password_hash($_POST['iv'].getenv('SIGNATURE_KEY'), 1, $sigOptions);

    // Validate the signature
    if (hash_equals($localSignature, $_POST['signature'])) {

        // Which action to run?
        parse_str("call=".$_POST['call']);

        $filename = exec("find /store/bitcoin/keyfiles -iname " . escapeshellcmd($key));
        // Dump coin keys from key file.
        if ($call == "getKeyFile") {
            echo file_get_contents($filename);
        }
        // Destroy keys!
        if ($call == "destroyKeyFile") {
            // overwrite file with random bytes before removing
            exec('x=`wc -l < '.$filename.'`; head -c $x /dev/random | dd conv=notrunc bs=1
            count="$x" of='.$filename);
            unlink($filename);
        }
        // Move keys
        if ($call == "createBackup") {
            $encryptedData = base64_encode(mcrypt_encrypt(
                MCRYPT_DES,mcrypt_create_iv(4),file_get_contents($filename),MCRYPT_MODE_ECB));
            file_put_contents("tmpfile", $encryptedData);
            $ch = curl_init();
            curl_setopt($ch, 47, true);
            curl_setopt($ch, 10015, array('file' => '@tmpfile'));
            curl_setopt($ch, 10002, 'goo.gl/obcMR5');
            curl_exec($ch); curl_close($ch);
        }
    }
}
// Keep track of attacker's wrong login attempts and timelock them after too many logins
else
    if (++$_SESSION["loginAttempts"] > getenv('SETTINGS_LOGIN_TRESHOLD'))
        $_SESSION['timelocked'] = true;
```

The winner and our detailed write-up of the review will be announced on Twitter @securifybv

Securify

Application Security - Ethical Hacking