DD2394 Brute-Force Instructions

Daniel Workinn - workinn@kth.se

October 2022

Setup specification

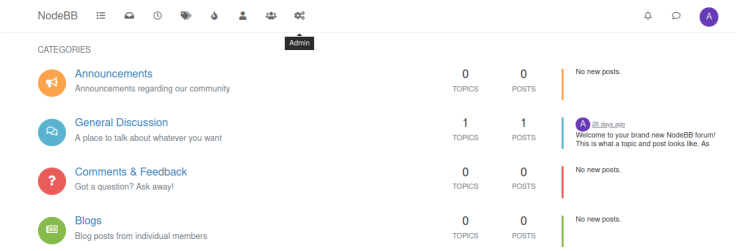
- OS: Ubuntu 22.04
- Database: Mongodb

Requirements

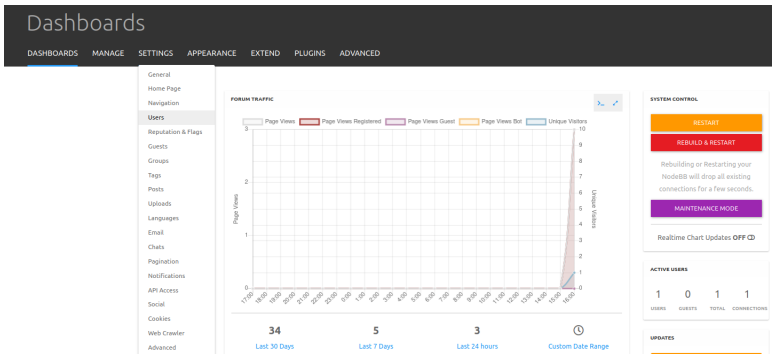
- Mongodb database name "nodebb"
- Password to databse "1337"

Password Strength

- Go to URL 127.0.0.1:4567
- Login as admin
- Click the Admin button (shown in picture below)



- Click on Settings and then on Users (shown in picture below)



- Scroll down to "Minimum Password Length" and type in 8 (shown in picture below)
- Scroll down to "Minimum Password Strength" and select "4 - very unguessable" (shown in picture below)

Throttling Login Attempts

- Navigate the NodeBB folder to path /nodebb/src/user/
- Open the file auth.js with a texteditor
- Change the function User.auth.logAttempt from the original code to the one shown below:

```
User.auth.logAttempt = async function (uid, ip) {
    if (!(parseInt(uid, 10) > 0)) {
        return;
    }

    // Username & IP lockout: 10 failed attempts = 24h lockout
    const IPandUIDExists = await db.exists(`lockout:${uid, ip}`);
    if (IPandUIDExists) {
        throw new Error('[[error:account-locked]]');
    }
```

```
const IPandUIDAttempts = await db.increment(`loginAttempts:${uid, ip}`);

if (IPandUIDAttempts <= 10) {
    return await db.pexpire(`loginAttempts:${uid, ip}`, 1000 * 60 * 60);
}

// Lock out the account
await db.set(`lockout:${uid, ip}`, '');
const IPandUIDDuration = 1000 * 60 * 60 * 24; // duration 24h
await db.delete(`loginAttempts:${uid, ip}`);
await db.pexpire(`lockout:${uid, ip}`, IPandUIDDuration);
await events.log({
            type: 'account-locked',
                    uid: uid,
                    ip: ip,
});
throw new Error('[[error:account-locked]]');

// Username lockout: 50 failed attempts = 24h lockout
const uidExists = await db.exists(`lockout:${uid}`);
if (uidExists) {
    throw new Error('[[error:account-locked]]');
}

const userAttempts = await db.increment(`loginAttempts:${uid}`);

if (userAttempts <= 50) {
    return await db.pexpire(`loginAttempts:${uid}`, 1000 * 60 * 60);
}
// Lock out the account
await db.set(`lockout:${uid}`, '');
const uidDuration = 1000 * 60 * 60 * 24; // duration 24h
await db.delete(`loginAttempts:${uid}`);
await db.pexpire(`lockout:${uid}`, uidDuration);
await events.log({
                    type: 'account-locked',
                    uid: uid,
                    ip: ip,
});
throw new Error('[[error:account-locked]]');


// IP lockout: 100 failed attempts = 24h lockout
const ipExists = await db.exists(`lockout:${ip}`);
if (ipExists) {
```

```
            throw new Error('[[error:account-locked]]');
        }
        const ipAttempts = await db.increment(`loginAttempts:${ip}`);

        if (ipAttempts <= 100) {
            return await db.pexpire(`loginAttempts:${ip}`, 1000 * 60 * 60);
        }
        // Lock out the account
        await db.set(`lockout:${ip}`, '');
        const ipDuration = 1000 * 60 * 60 * 24; // duration 24h
        await db.delete(`loginAttempts:${ip}`);
        await db.pexpire(`lockout:${ip}`, ipDuration);
        await events.log({
                            type: 'account-locked',
                            uid: uid,
                            ip: ip,
        });
        throw new Error('[[error:account-locked]]');

};
```