FORGOTTEN THREAT - FORK BOMB

First off lets get the basics sorted first, the threat of a fork bomb can be classified as a Denial-of-Service attack, as it will totally exhaust resources on the server being victimized. But, it can only be made possible if the server has not been configured fully - most especially the limits placed upon the users of the system. I know that each possible classification has its advocates, but no matter what you think the threat is real.

What Is It?
The threat is simply that any user able to execute commands on a system can bring the system to a grinding halt by using the resources. And you did not read wrong, anyone can do it. It does not require any major technical knowledge, in fact here it is here; as a simple shell script..

```
$0 &
exec $0
```

Do not test this on any server you cannot reboot or which is being used for production purposes. What the above script does is just create multiple instances of itself and just carrys on doing it until the CPU just grinds to a halt, and if the CPU does not give out, then the memory or some other resource will. And you can get nasty as well..

```
echo "oh darn it" >> fork.$RANDOM &
$0 &
exec $0
```

This will not only create multiple instances of the process, but each process will create a file with a short message inside. Thus adding a new threat to the disk space of the targeted system. In fact you could add other attacks in as well.

What To Do About It?
Well, as I started off saying, this threat is made possible if the server has not been fully configured. You see, what you can do is use the *ulimit* command to limit the number of possible processes a user can have. Add the following -purely a suggested limit- to the startup files (*/etc/profile*) of the users.

```
ulimit -u 200
```

Or you can do the following to apply the limit to all users upon system startup

```
echo "* hard nproc 200" >> /etc/security/limits.conf
```

As far as threat of disk usage goes, the only thing you can do is to have the system files and the user directories - including the */tmp* folder- seperate. And it is actually as simple as that, so there is really no reason to not have it set on your systems.

Final Words
Have fun trying it, lets face it, there is nothing like actually typing a couple of commands and bringing a server to a grinding halt. Just bear in mind that I warned you about testing it on production machines and defintely on machines you do not have responsibility for. As always, learn, have fun and make your systems secure.