

## Proxy Protocol Receiver

### Use Case:

Remove the PROXY protocol header from TCP payload before forwarding the request to backend.

### F5 iRules:

```
#PROXY Protocol Receiver iRule
# c.jenison at f5.com (Chad Jenison)
when CLIENT_ACCEPTED {
    TCP::collect
}

when CLIENT_DATA {
    if {[TCP::payload 0 5] eq "PROXY"} {
        log local0. "Proxy Protocol Header found"
        set proxy_string [TCP::payload]
        set proxy_string_length [string length [TCP::payload]]
        scan $proxy_string {PROXY%%s%%s%%s%%s} tcpver srcaddr dstaddr srcport ds
tport
        log local0. "TCPVER: $tcpver - SRCIP: $srcaddr - DSTIP: $dstaddr - SRC
PORT: $srcport - DSTPORT: $dstport"
        TCP::payload replace 0 $proxy_string_length ""
    } else {
        #uncomment below lines if you want to drop connections that are missin
g proxy protocol line
        #reject
        #log local0. "No Proxy Header found - Connection Rejected from [IP::cl
ient_addr]"
    }
    TCP::release
}
```

URL: <https://devcentral.f5.com/codeshare/proxy-protocol-receiver>

## NetScaler Solution:

```
add policy expression tcp_ver
CLIENT.TCP.PAYLOAD(107).AFTER_STR("PROXY").AFTER_STR("
").BEFORE_STR(" ")
```

```
add policy expression src_ip
CLIENT.TCP.PAYLOAD(107).AFTER_STR("PROXY").AFTER_STR("
").AFTER_STR(" ").BEFORE_STR(" ")
```

```
add policy expression dst_ip
CLIENT.TCP.PAYLOAD(107).AFTER_STR("PROXY").AFTER_STR("
").AFTER_STR(" ").AFTER_STR(" ").BEFORE_STR(" ")
```

```
add policy expression src_port
CLIENT.TCP.PAYLOAD(107).AFTER_STR("PROXY").AFTER_STR("
").AFTER_STR(" ").AFTER_STR(" ").AFTER_STR(" ").BEFORE_STR(" ")
```

```
add policy expression dst_port
CLIENT.TCP.PAYLOAD(107).AFTER_STR("PROXY").AFTER_STR("
").AFTER_STR(" ").AFTER_STR(" ").AFTER_STR(" ").AFTER_STR("
").BEFORE_STR(" ")
```

```
set syslogParams -UserDefinedAuditlog yes
```

```
add auditmessageaction log_proxy INFORMATIONAL '"PROXY Protocol
Detected. TCP Version - "+tcp_ver+" SourceIP - "+src_ip+" DestIP -
"+dst_ip+" SourcePort - "+src_port+" DestPort - "+dst_port'
```

```
add rewrite action delete_proxy delete
CLIENT.TCP.PAYLOAD(107).BEFORE_STR("\r\n")
```

```
add rewrite policy check_proxy CLIENT.TCP.PAYLOAD(5).EQ("PROXY")
delete_proxy -logAction log_proxy
```

Bind Rewrite policy to a TCP Type vserver. Here the expressions define the TCP payload parsing syntax for finding out PROXY protocol header. Once found the Rewrite action is going to delete it from the TCP payload in request.