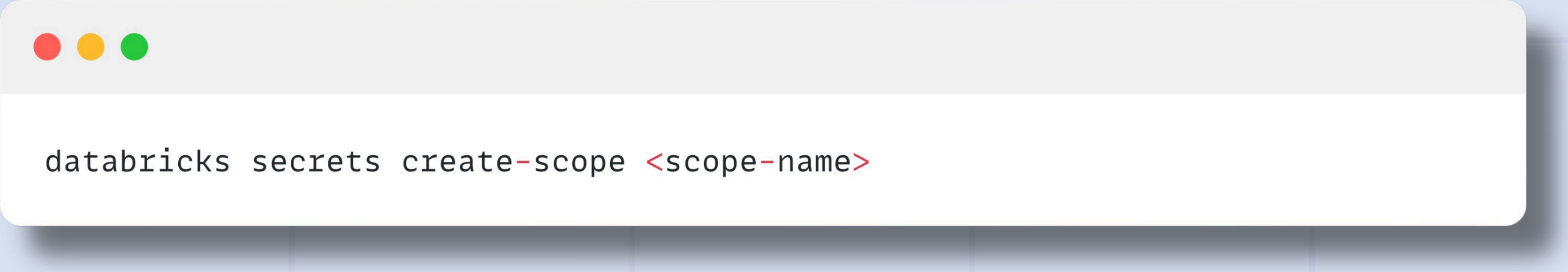
meshynix

Complete Secret Management in Databricks

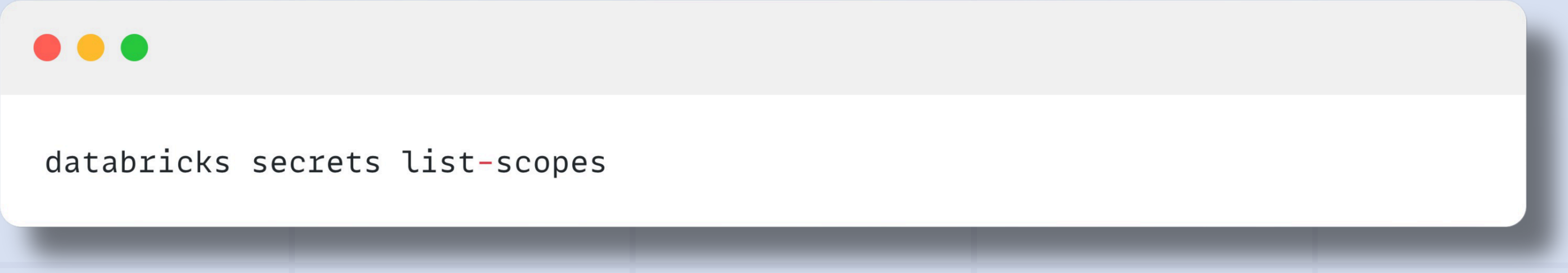
Managing Secrets via CLI

Create a Secret Scope:



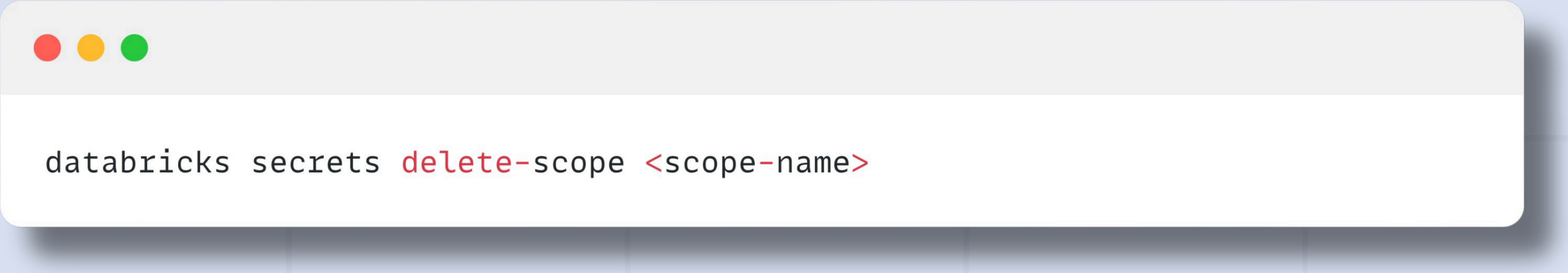
```
databricks secrets create-scope <scope-name>
```

List Secret Scopes:



```
databricks secrets list-scopes
```

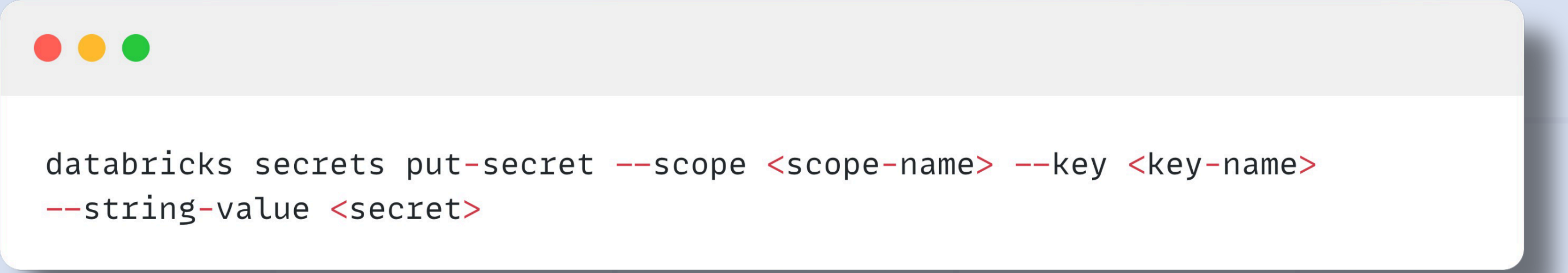
Delete a Secret Scope:



```
databricks secrets delete-scope <scope-name>
```

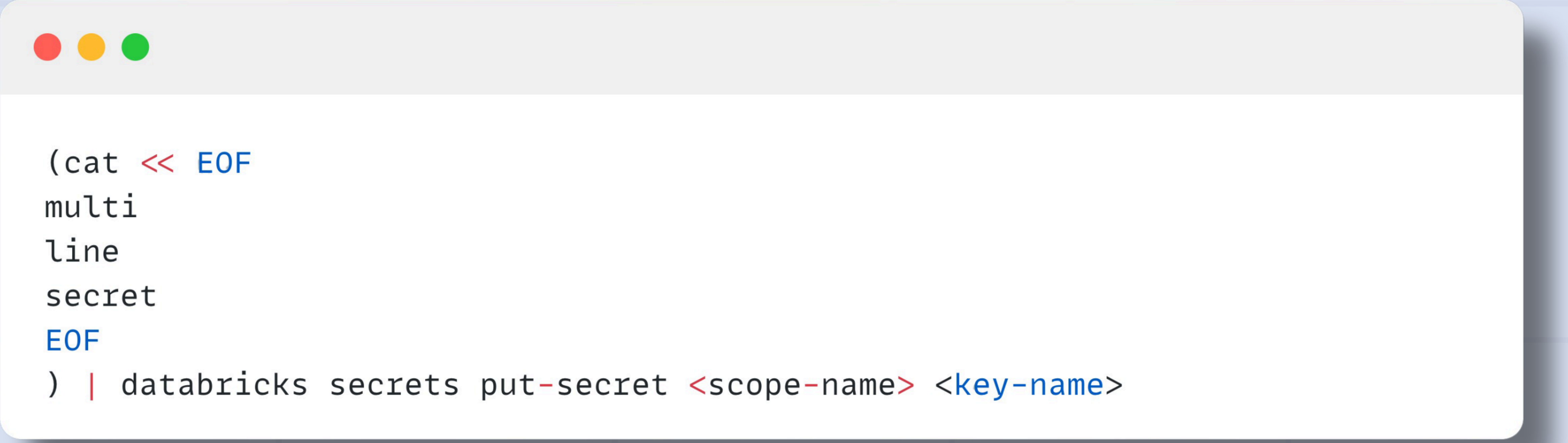
Ways to add a Secret

Single-line secret via CLI:



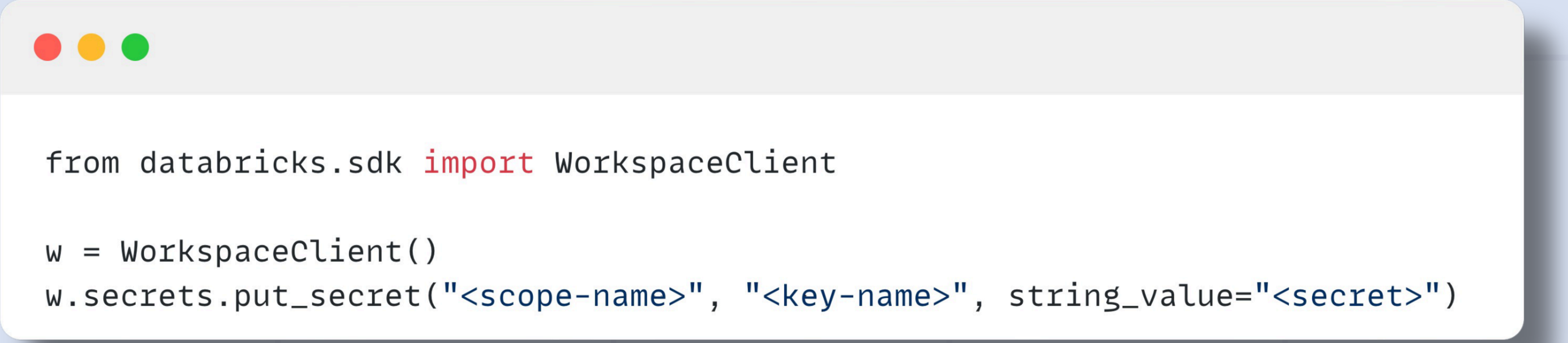
```
databricks secrets put-secret --scope <scope-name> --key <key-name>
--string-value <secret>
```

Multi-line secret via stdin:



```
(cat << EOF
multi
line
secret
EOF
) | databricks secrets put-secret <scope-name> <key-name>
```

Using Python SDK:

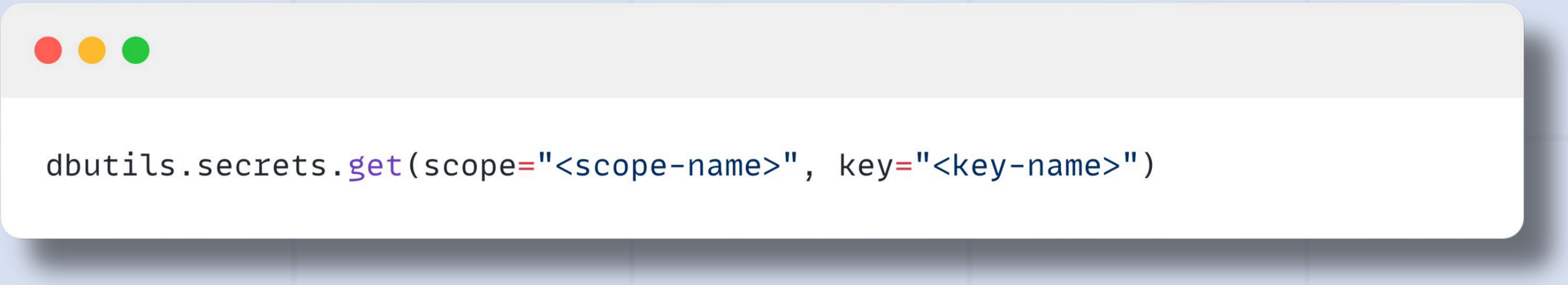


```
from databricks.sdk import WorkspaceClient

w = WorkspaceClient()
w.secrets.put_secret("<scope-name>", "<key-name>", string_value="<secret>")
```

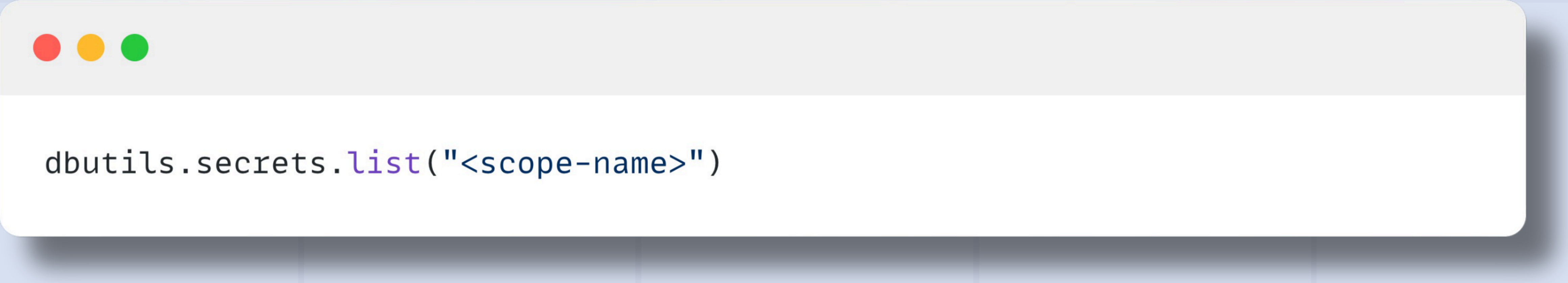
Read & List Secrets

Read a secret in a notebook:



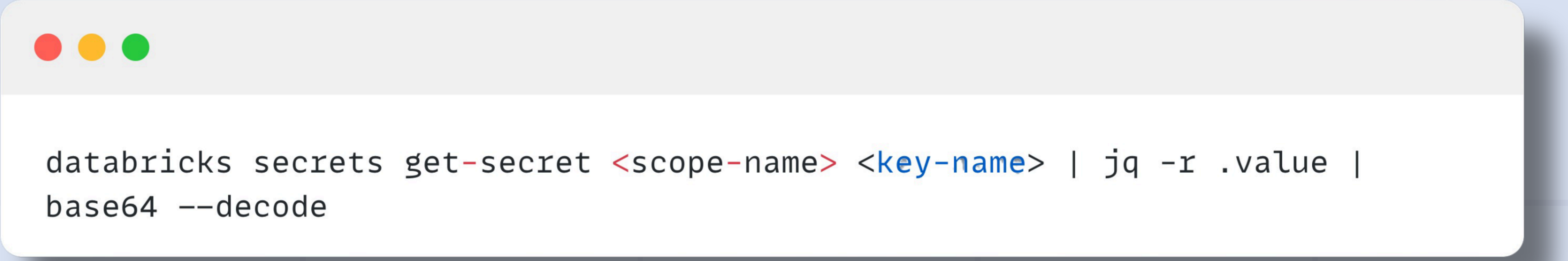
```
dbutils.secrets.get(scope="<scope-name>", key="<key-name>")
```

List secrets in a scope:



```
dbutils.secrets.list("<scope-name>")
```

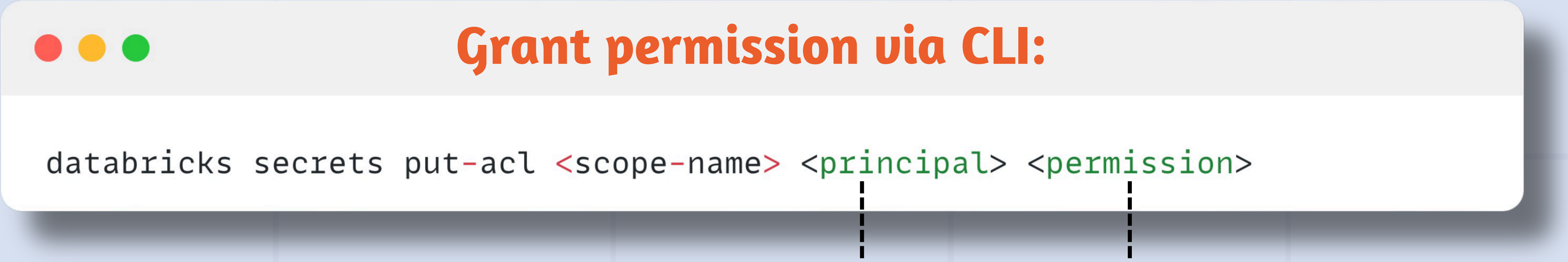
CLI (with decoding):



```
databricks secrets get-secret <scope-name> <key-name> | jq -r .value |
base64 --decode
```

Managing Access & Best Practices

ASSIGNING PERMISSIONS:



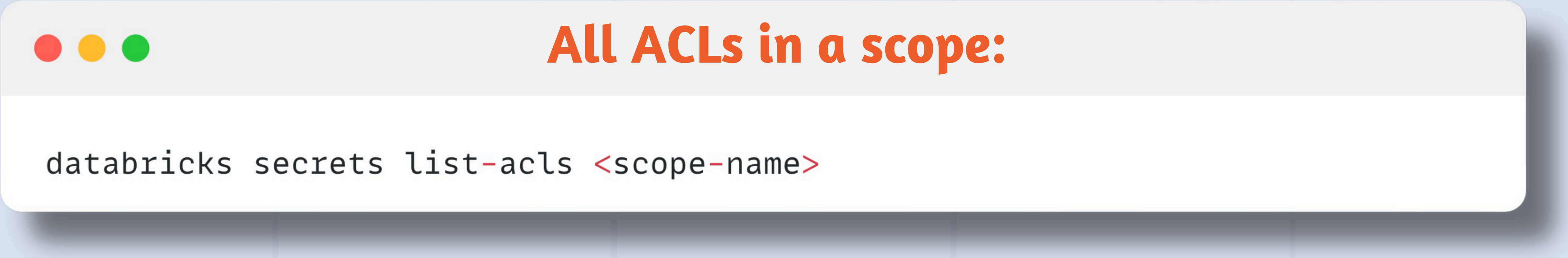
Grant permission via CLI:

```
databricks secrets put-acl <scope-name> <principal> <permission>
```

User email, group name, or service principal ID.

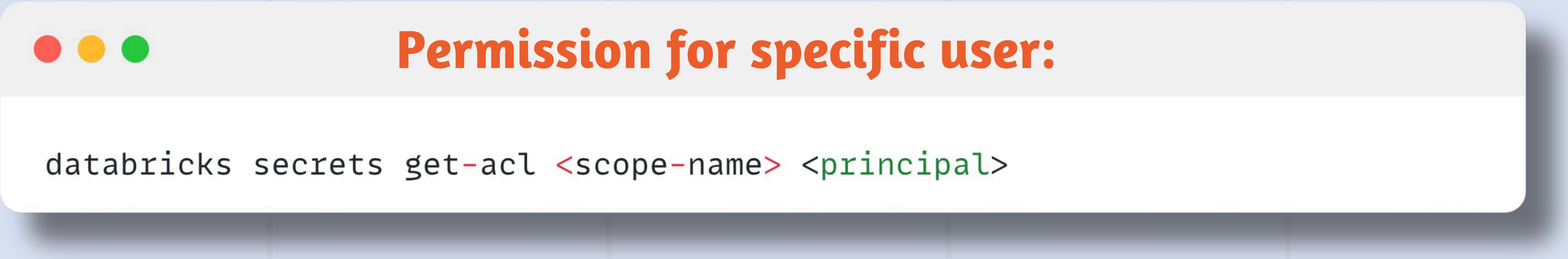
`READ`, `WRITE`, or `MANAGE`

VIEW PERMISSIONS:



All ACLs in a scope:

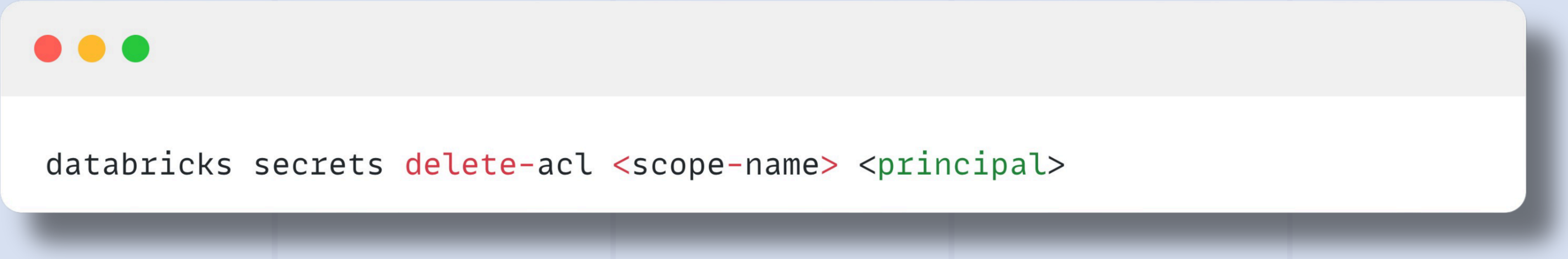
```
databricks secrets list-acls <scope-name>
```



Permission for specific user:

```
databricks secrets get-acl <scope-name> <principal>
```

REVOKE PERMISSIONS:



```
databricks secrets delete-acl <scope-name> <principal>
```