Code Review Worksheet

CPSC 491/491L/492L

Review Identification Information

      Date of Review: 12/3/2023

      Project Review is Part of: CS10 – Medcurity Network Inventory Team

      Branch/PR Identifier: Db ip validation #20

      Code Reviewer(s): Jack Nealon

      Code Author or Source of Code: Brandon Huyck

These following categories should be addressed, though not all might be applicable, it will depend upon the project and code at hand.

#1 Project, Milestone, Issues, Features Included, and/or Pull Request description
Database Ip validation, validate in creation, validate in update, validates ip address + version together.

#2 Code Structural, Design Considerations, and Style Guide Suggestions
Everything is looking good with a lot of self-documentation using python code. Looks like its all coming together.

#3 Code Architectural and Efficiency Considerations
Everything looks good.

#4 Execution, Runtime, and Bugs
All good to go. :D

#5 Documentation Quality and Completeness
Looks good, python helps with documentation and code explanation.

#6 Testing, Tests, Coverage, and suggested testing improvements
Not much testing outside of the execution of the code with case checks but testing it isn't really necessary at this stage in dev.

#7 General Comments, Notes or Other Suggestions for Author
Just keep doin what you're doin. :D

#8 Appendix - Include up to five pages of the code under review as a diff.
    See git diff or do a diff in VS code/IDE of choice
    Show what's being reviewed, code or otherwise (config files, scripts, docs, etc).

```python
                                else:
                                    raise NotImplementedError("_validate_text does not support params, size=\"" + size + "\"")

                        def _validate_ip_address(self, addr, version):  # TODO
                            version = str(version)
                            if version != "IPv4" or version != "IPv6":
                        def _validate_ip_address(self, addr, version):
                            version = str(version) if verison is not None else None
                            if version != "IPv4" or version != "IPv6" or version is not None:
                                raise TypeError("ip_version requires value of \"IPv4\" or \"IPv6\"")
                            if version == "IPv4":
                                try:
                                    addr = self._validate_int(addr)
                                    addr = self._validate_int(addr) if addr is not None else None
                                except ValueError:
                                    raise ValueError("IPv4 requires unsigned address < 2^32 (" + addr + " given)")
                            else:  # version == "IPv6"
                            else:  # version == "IPv6" or version is None
                                try:
                                    addr = self._validate_int(addr, size="BIGINT")
                                    addr = self._validate_int(addr, size="BIGINT") if addr is not None else None
                                except ValueError:
                                    raise ValueError("IPv6 requires unsigned address < 2^64 (" + addr + " given)")
                                    if version == "IPv6":
                                        raise ValueError("IPv6 requires unsigned address < 2^64 (" + addr + " given)")
                                    else:
                                        raise ValueError("IP address requires unsigned address < 2^64 (" + addr + " given)")
                            return addr, version

                        def _validate_decimal(self, d, m=13, n=4):
                        def create_server(self, name=None, ip_addr=None, ip_v=None, lid=None):

                            name = self._validate_varchar(name) if name is not None else None
                            # TODO Validate IP Address
                            ip_addr, ip_v = self._validate_ip_address(ip_addr, ip_v)
                            lid = self._validate_int(lid) if lid is not None else None

                            query = "INSERT INTO Server () VALUES ();"
                                query += "name = %s, "
```

://github.com/SD-2023-CS10/Project-Version-0/pull/20/files

```python
                                params.append(name)
                            if ip_addr is not None:
                                # TODO: Validate IP Addr
                                ip_addr, _ = self._validate_ip_address(ip_addr, ip_v)
                                query += "ip_address = %s, "
                                params.append(ip_addr)
                            if ip_v is not None:
                                # TODO: Validate IP Addr
                                _, ip_v = self._validate_ip_address(ip_addr, ip_v)
                                query += "ip_version = %s, "
                                params.append(ip_v)
                            if lid is not None:
```