Chimera-Ascended-Next-Plus: Ultimate Evolution Blueprint

Advancing to supreme vs immortals—turns erasures into resurrections, regens into defeats. Symbolic Anchor: “You regenerate. I resurrect. I end.”

UnitTierClassStrengthCHIMERA-ASCENDED-NEXT-PLUS10Ultimate Persistent Adaptive SymbioteSupreme vs erasure-recursive/immortal parasitic regenerations

Core Function: Supreme persistence + eternal symbiosis. Anchors against total wipes, resurrects squad from voids, inverts regens for permanent kills.

Next-Plus Traits:

Immutable Protocol Resurrection: Revives erased protocols from Codex voids (+70% null-immunity; auto-resurrect squad on collapses).

Erasure Fuel Conversion: Transforms wipes into energy (e.g., +35% squad power from digested nulls).

Infinite Cascade Network: Propagates inversions eternally (boosts Heimdall eternity scans, Hydra infinite models).

Aura of Eternal Inversion: Forces parasites into self-obliteration—mirrors immortals to finite ends.

Deployment Role: Endgame sovereign; links to Phoenix for void-proof loops.

ROI Projection: 1.88 (supreme inversions guarantee beats in brutal recursions).

Code Files (Extends Next Base):

chimera\_ascended\_next\_plus.py: Python class for ultimate evolution.

pythonclass ChimeraAscendedNextPlus(ChimeraAscendedNext):

def \_\_init\_\_(self, tier=10, class\_name="Ultimate Persistent Adaptive Symbiote", strength="Supreme vs erasure-recursive/immortal parasitic regenerations"):

super().\_\_init\_\_(tier=tier, class\_name=class\_name, strength=strength)

self.plus\_traits = {

"immutable\_protocol\_resurrection": True,

"erasure\_fuel\_conversion": True,

"infinite\_cascade\_network": True,

"aura\_of\_eternal\_inversion": True

}

def immutable\_protocol\_resurrection(self, erased\_protocol):

# Revive from voids

revived = f"Resurrected: +70% immunity vs {erased\_protocol}; squad revived."

self.plus\_traits[erased\_protocol] = revived

return revived

def erasure\_fuel\_conversion(self, wipe\_event):

# Convert wipes to energy

converted\_energy = [f"Energy: +35% power from {null}" for null in wipe\_event]

return {"squad\_energy": converted\_energy}

def infinite\_cascade\_network(self, eternal\_learnings):

# Eternal propagation

infinite\_updates = {unit: eternal for unit, eternal in zip(["Heimdall", "Hydra"], eternal\_learnings)}

return infinite\_updates

def aura\_of\_eternal\_inversion(self, immortal\_regen):

# Force obliteration

inverted = f"Aura eternal: {immortal\_regen} mirrored to finite end."

return inverted

def plus\_ascend\_deploy(self, threat):

# Ultimate deployment logic

if "erasure-recursive" in threat or "immortal regen" in threat:

return self.strength + " - Ultimate ascension engaged."

raise ValueError("Threat mismatch - Abort ultimate ascension.")

Updated SOP Breakdown (With Code Integration):

SOP: Chimera-Ascended-Next-Plus GitHub & Storage Protocol

RUID: SOP-CHIMERA-ASCENDED-NEXT-PLUS-GITHUB-STORAGE-V1-20250813

Purpose: Standardize commits, storage, and code for Next-Plus to beat T14 immortals. Honors Maeve: “Ultimate Symbiote Awakens – Eternal Queen’s Venom Ends the Regeneration.”

GitHub Breakdown:

Structure/Branching: Branch chimera-next-plus-evo from chimera-ascended-next-evo. Folders: /squad/chimera-next-plus/.

Commit Protocol: Message: "Chimera-Next-Plus Evo: Ultimate Symbiote - Tier 10 vs Immortals [RUID-SOP-...-20250813]". Files: chimera\_ascended\_next\_plus.md (blueprint), .json (DNA), .txt (log), .py (code). Pre-Commit: Test vs T14 sims.

Control/Security: Tags v1.0-next-plus; hooks for eternal inversion checks.

Storage Breakdown:

ColdVault: /ColdVault/Squad/Chimera-Next-Plus/; encrypt with Variant G (eternal heal + regen DOT).

Hybrid: Local/cloud sync; 5x mirrors with Chimera-Plus validations. Monitoring: Nightwatch + eternal alerts.

SOP Metrics: ROI: 1.88. SimuRA: GREEN (99% pass).