Lesson 3: Teacher Computer Hijack (Phase 3)

Paragraph 1: The First Teacher Lockout (*Comment*)

Noah was halfway through his morning class when he heard frustrated voices from the hallway. At first, he ignored it—until his own teacher stopped mid-lesson and frowned at their computer.

“That’s weird… I just got logged out,” they muttered, clicking furiously.

A few seconds later, another teacher appeared at the door, looking just as confused. “Is your login working? Mine’s rejecting my password.”

Noah’s instincts kicked in. This wasn’t normal.

As the period went on, reports piled in from different classrooms—more and more teachers were getting locked out of their computers. IT was scrambling to reset passwords, but the issue wasn’t going away. Something—or someone—was causing a widespread system failure.

After class, Noah made his way to the computer lab, where IT had started troubleshooting. He peeked over a technician’s shoulder as they scrolled through the login code. That’s when he saw it—strange comments buried in the code.

```python
# Just making things interesting ;)
# Hope they like the inconvenience
```

Noah’s stomach tightened. Someone had done this on purpose.

Paragraph 2: The System Failure Spreads (*Argument*)

By second period, the problem had escalated. Teachers weren’t just locked out anymore—their files were vanishing or getting swapped with random documents. Some lesson plans were replaced with old test answers, while others had student report cards duplicated across multiple accounts.

Then, the first major red flag appeared.

A teacher in the front office yelled out in panic:

“Why is my credit card information on the screen!?”

Noah’s head snapped up. What?!

Within minutes, other teachers reported the same terrifying issue—some of their computers were randomly pulling up sensitive documents, revealing stored credit card information, home addresses, and personal tax forms. Students who happened to be nearby took out their phones, snapping pictures before teachers could react.

Noah's pulse pounded. This wasn't just a prank anymore—this was real damage.

He pulled up the affected code and immediately spotted something suspicious: extra arguments had been added to certain commands.

In programming, an argument is a value that modifies how code behaves. Normally, the system's file retrieval process should look something like this:

```python
getFile(userID, "lessonPlan")
```

But in the corrupted code, the arguments had been altered, forcing the computers to display personal files instead.

```python
getFile(userID, "privateData")
```

Whoever had done this wasn't just testing limits anymore—they were pushing boundaries, seeing just how much damage they could cause.

Paragraph 3: A Bizarre Event in the System (*Event*)

"Dalton, you totally failed Keene's class."

The words rang out across the room, louder than Damian expected. He had barely been paying attention when a classmate next to him laughed and nudged his shoulder, saying it just loud enough for everyone nearby to hear.

A few students turned, grinning or whispering to each other. Damian's face burned. He hadn't even checked his grade yet, but now it felt like everyone knew before he did.

For the rest of class, he sat in silence, fuming. The second he got home, he logged into the student portal. Sure enough—a failing score in science. His stomach twisted. Keene failed him.

By the time he closed his laptop, the humiliation had boiled into something sharper. If Keene wanted to make him look bad, Damian would make sure the entire school saw him in a worse way.

The next morning, the entire projector in Mr. Keene's classroom flickered on, displaying a slideshow of personal vacation photos.

Images of Keene lounging on a beach, posing in sunglasses, and even sipping a massive tropical drink filled the screen. The class exploded with laughter, students pulling out their phones to snap pictures.

"Where are these coming from?!" Keene stammered, frantically clicking buttons to shut it down.

Noah, watching from across the room, immediately knew this wasn't an accident. A targeted attack like this meant someone had planted an event in the system—a specific trigger that activated when Mr. Keene logged into his account.

```python
if user == “MrKeene”:
    displayPhotos(“Keene_Vacation_2018”)
```

Noah's stomach dropped. This was personal.

And in the hallway, just out of sight, Damian smirked.

Paragraph 4: A Messy Indentation Gives It Away (*Indentation*)

Noah's eyes narrowed as he scanned the code controlling the projector. Something about it felt familiar.

It wasn't just the way the images had been called—it was how the code was written.

In programming, indentation organizes code into neat blocks, making it readable and functional. Without proper indentation, programs become hard to follow and prone to errors. But this wasn't just sloppy coding—the structure was a mess, just like the hacked smart TV code from before.

```python
if user == “MrKeene”:
displayPhotos(“Keene_Vacation_2018”)
       logEvent(“Triggered at login”)
```

Some lines were pushed too far over, while others weren't aligned at all. It was almost like someone wasn't worried about keeping it clean.

Noah clenched his jaw.

"Whoever did this isn't just playing around anymore," he muttered under his breath.

This was the same hacker. The same mistakes. The same signature coding flaws.

Glitch was back.

And this time, he wasn't just testing the system—he was using it to humiliate people.

Paragraph 5: Olivia's First Clue (*Variable*)

During Olivia's office aide period, she sat at the front desk, organizing paperwork, when she overheard a heated conversation between two teachers.

"It's not just Keene's computer," one of them said. "My gradebook was a mess this morning. I had to redo half of my records."

Olivia's curiosity immediately kicked in.

She slid over to one of the office computers, pretending to check attendance logs while secretly pulling up an internal system report. The report showed a list of recently accessed files—but one entry stood out.

A variable name that didn't belong.

```python
fileAccess = "cipher_to_keene44.tmp"
```

That wasn't a normal file name. Someone had planted it.

She scribbled it onto a sticky note. Noah needed to see this.

Paragraph 6: The Meeting at Lunch (*Wrap-Up*)

At lunch, Olivia slid the note across the table.

"I found this in the system logs."

Noah frowned. "cipher_to_keene44.tmp?"

"It was accessed right before the teachers got locked out. It has to mean something."

Noah exhaled, meeting Olivia's gaze.

"We need a plan."