```lua
local ReplicatedStorage = game:GetService("ReplicatedStorage")

local terrain = workspace.Terrain

local volcanoEvent = ReplicatedStorage:WaitForChild("VolcanoEvent")

local function generateVolcano()
    local basePosition = Vector3.new(0, 0, 0)
    local height = 80
    local baseRadius = 60
    local topRadius = 20
    local craterRadius = 18
    local lavaRadius = 14
    local steps = 30

    local stepHeight = height / steps
    local radiusStep = (baseRadius - topRadius) / steps
    local lavaStartStep = math.floor(steps * 0.75) -- Top 25%

    -- Step 1: Build volcano base and sides
    for i = 0, steps - 1 do
        local currentHeight = i * stepHeight
        local currentRadius = baseRadius - (i * radiusStep)
        local position = basePosition + Vector3.new(0, currentHeight, 0)

        local size = Vector3.new(currentRadius * 2, stepHeight, currentRadius * 2)
        local cframe = CFrame.new(position)

        local material
        if i >= lavaStartStep and i < steps - 2 then
            material = Enum.Material.CrackedLava
        else
            material = Enum.Material.Rock
        end

        terrain:FillBlock(cframe, size, material)
    end
    local craterCenter = basePosition + Vector3.new(0, height + 4, 0)
    terrain:FillBall(craterCenter, craterRadius, Enum.Material.Air)
    local lavaCenter = craterCenter + Vector3.new(0, -2, 0)
    terrain:FillBall(lavaCenter, lavaRadius, Enum.Material.CrackedLava)

    for i = 1, 6 do
        local offsetX = math.random(-4, 4)
        local offsetZ = math.random(-4, 4)
        local drop = lavaCenter + Vector3.new(offsetX, -i * 6, offsetZ)
        terrain:FillBall(drop, 3, Enum.Material.CrackedLava)
    end
end

volcanoEvent.OnServerEvent:Connect(function(player)
    generateVolcano()
end)
```